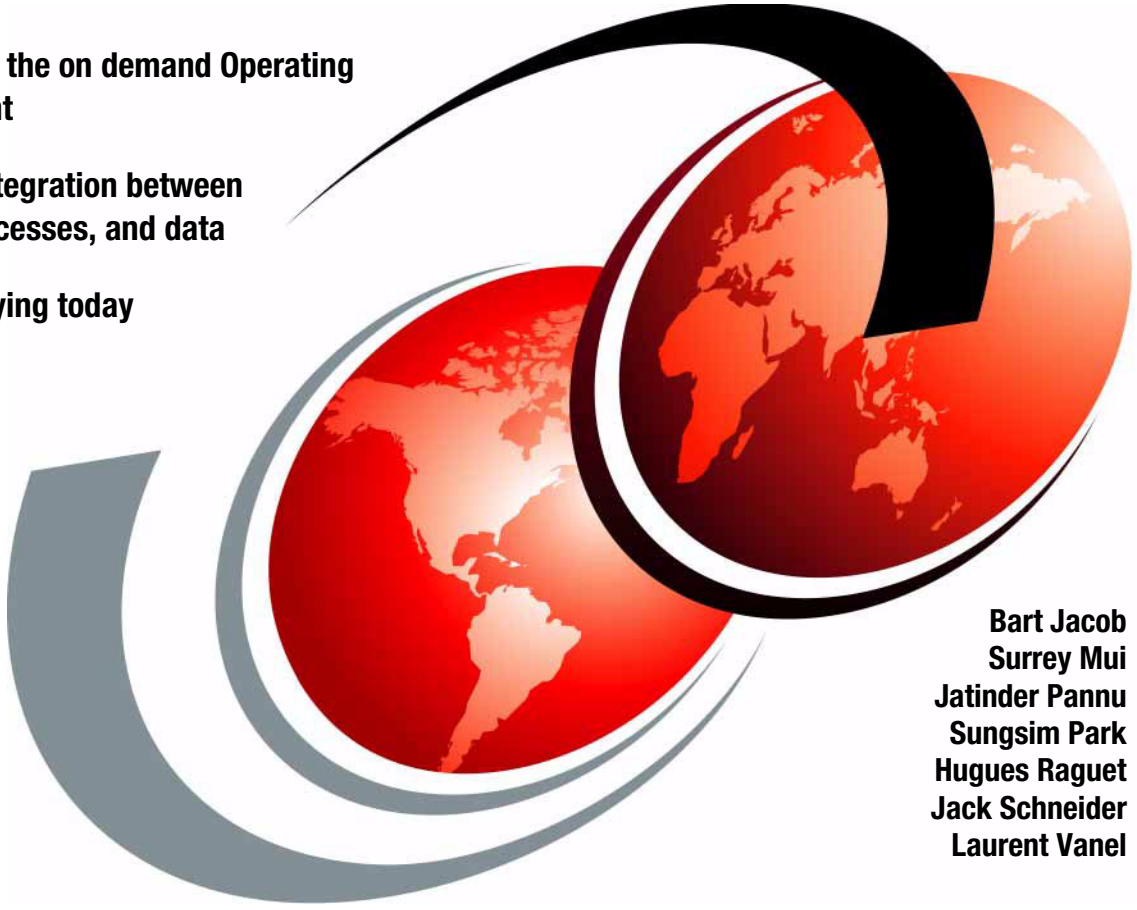
IBM

On demand Operating Environment: Creating Business Flexibility

- Introducing the on demand Operating Environment
- Enabling integration between people, processes, and data
- Start deploying today



Bart Jacob
Surrey Mui
Jatinder Pannu
Sungsim Park
Hugues Raguet
Jack Schneider
Laurent Vanel

Redbooks

ibm.com/redbooks

International Technical Support Organization

On demand Operating Environment: Creating Business Flexibility

April 2004

SG24-6633-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xi.

First Edition (April 2004)

Contents

Figures

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

@server®	Informix®	Redbooks™
Redbooks (logo)  ™	IBM®	S/390®
ibm.com®	Lotus®	Sametime®
iSeries™	MQSeries®	Tivoli®
pSeries®	OS/390®	WebSphere®
AS/400®	Rational Developer Network®	Workplace Messaging™
CICS®	Rational Rose®	XDE™
DB2®	Rational®	

The following terms are trademarks of other companies:

Microsoft, Windows the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Preface

This redbook (along with its companion volume, *On demand Operating Environment: Managing the Infrastructure*, SG24-6634), provides an insight into the kind of Operating Environment required to support an on demand business.

It provides an overview of the architecture of an on demand Operating Environment and describes in more detail the components that are required to create business flexibility through integration. To meet the business needs of being responsive, variable, focused and resilient, an on demand Operating Environment must be integrated, autonomic, virtualized and open. Though these attributes are all interrelated, this redbook focuses on the integration component as the key enabler of business flexibility.

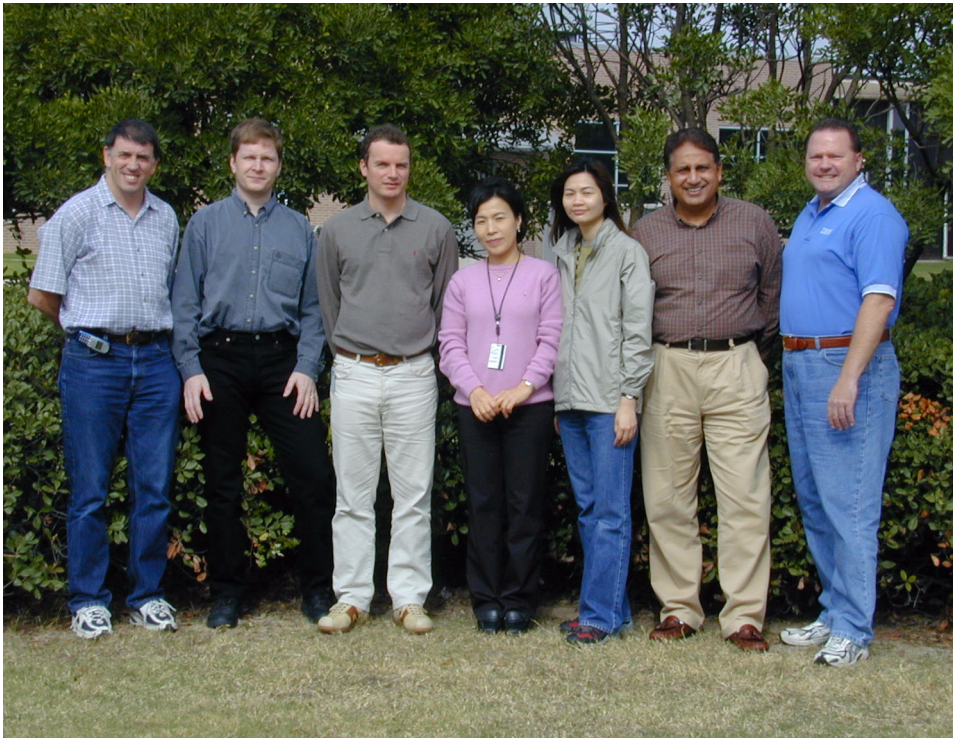
A complete on demand Operating Environment is a vision and a goal that many enterprises aspire to reach. However, it is not something that will be attained overnight or by installing a specific set of products. It is something that will be reached through a step-wise progression.

This redbook provides descriptions of several approaches or how tos that one can choose to start implementing pieces of an on demand Operating Environment today. Which approach is right for the reader will depend on their specific business environment and their immediate needs.

Our objective is to help the reader better understand what an on demand Operating Environment is and how they can take steps today to start putting one in place. This redbook does not go into detailed implementation plans for each technology or product it references, but rather provides a level of information sufficient for the reader to start building a strategy and architecture best suited for their needs. Product specific details can be obtained from product documentation and product related redbooks.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.



From left to right: Bart Jacob, Laurent Vanel, Hugues Raguet, Sungsim Park, Surrey Mui, Jatinder Pannu, Jack Schneider

Bart Jacob is a Senior Consulting IT Specialist in the IBM® International Technical Support Organization, Austin Center. He has 23 years of experience providing technical support across a variety of IBM products and technologies, including communications, object-oriented software development, and systems management. He has over 10 years of experience at the ITSO, where he has been writing IBM Redbooks™ and creating and teaching workshops around the world on a variety of topics. He holds a Master's degree in Numerical Analysis from Syracuse University.

Surrey Mui is an Advisory IT Specialist at IBM Global Services, Hong Kong. She has over ten years of experience in the Information Technology industry. Before joining IGS-HK in August 2003, she was a member of the Advanced Customer Engineering team working in the Tivoli® Security Business Unit in Australia (part of the IBM Software Group). Her area of expertise is in providing architecture and integration solutions for customers using the Tivoli Access Manager product suite. This includes building specialized development modules for customers based on the Access Manager product suite. Surrey was a member of the

DASCOM team when they were acquired by IBM. Surrey holds a degree in Business Computing from Griffith University, Australia.

Jatinder Pannu is a Senior Consulting Client IT Architect in IBM US. He has over twenty years of experience in the IT industry, and has worked for IBM for nineteen years planning requirements and designing systems with customers. He holds a Master's Degree in Computer Science from Syracuse University and an MBA from Duke University. His areas of expertise include designing e-business systems.

Sungsim Park is a Consulting IT Specialist and Solution Assurance manager in Technical Sales Support in IBM Korea. She has over 19 years of experience working on iSeries™, e-business applications and patterns, and Linux. She has taught several e-business on demand classes to IBM employees and Business Partners in Korea.

Hugues Raguet is an IT Architect in e-business solutions Technical Sales Support at IBM France. He has 9 years of experience in the IT industry. He holds a degree in electronic engineering from the ENSEEIHT University in Toulouse. His areas of expertise include the design of e-business solutions.

Jack Schneider is a Senior Consulting IT Architect in IBM US. He joined IBM in 1987 and has a consulting background with over 25 years of experience in architecting, designing, and implementing leading edge technology solutions to address customer's complex business needs.

Laurent Vanel is a System Architect in IBM France. He started as an IT specialist in the pSeries® brand in 1990. His IBM experience included a three year assignment at the Austin ITSO. In his current position as a System Architect, Laurent is technically responsible for the infrastructure proposed to a selected set of large customers.

Thanks to the following people from across IBM who assisted with the development of this redbook:

Alistair Rennie
Angel Luis Diaz
Annette Miller
Bala Rajaraman
Ben Amaba
Bruce Benfield
Connie Nelin
Dan Wolfson
Elisabeth Poigin
Errol R Denger
Fabio L Marras

Fredrik Carlegren
Georgina Castanon
Howie Miller
Jesus R Grana
Joseph Labriola
Julie Schuneman
Madeline Nick
Marc-Thomas Schmidt
Marjorie Macintyre
Mark Cathcart
Mark Palmer
Matthew P Haynos
Miles Barel
Pascal Durazzi
Ralf Heindoerfer
Randy B Daniel
Robert Liburdi
Robert Spory Jr
Roy Bowen
Simon K Johnston
IBM Learning Technologies at a Glance team

Our apologies to any others that may have been inadvertently left off this list. Your help was appreciated.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. JN9B Building 003 Internal Zip 2834
  11400 Burnet Road
  Austin, Texas 78758-3493

Part 1

Overview

In this part we introduce the on demand Operating Environment framework and the components that provide integration between people, processes, and information.

1

Introduction

As enterprises strive to meet their current challenges, some of which are generally described in this section, they require an IT infrastructure that supports their business goals. This redbook and its companion (*On demand Operating Environment: Managing the Infrastructure*, SG24-6634), describe an IT infrastructure that enables business to be more responsive, variable, focused, and resilient. Enterprises that exhibit these four attributes are what IBM calls *on demand* businesses. The IT infrastructure described here and that supports on demand businesses is called an *on demand Operating Environment*.

In these redbooks we provide an overview of the components of an on demand Operating Environment and show how enterprises can start putting pieces in place today that address immediate business needs. Over the longer term, enterprises can build on these initial steps to evolve their IT infrastructure to an on demand Operating Environment that enables the business to focus on their core competencies and meet whatever new challenges may arise.

1.1 Getting to on demand

On demand is not about technology for the sake of technology, it's about enabling new ways of doing business. It's about helping an organization reach new levels of innovation while continuing to deliver the improvements in productivity necessary to improve the bottom line. Yet the underlying technology makes an on demand business fundamentally different.

When business processes have been integrated end-to-end; across a company and with its key partners, suppliers, and customers; it has the ability to respond to any customer demand, market opportunity, or external threat. Yet, there's a lot of work to be done. Today's infrastructure is complex and rigid. Because much of it is based on proprietary hardware and software, delivered well before industry standards were established, it's difficult to make all the pieces work together. It's even more challenging to make them deliver the flexibility necessary to support today's dynamic business environment.

The need for change is forcing the emergence of a new computing model. This new, on demand model blends the robust nature of the traditional IT computing model with the industry-standards-based computing model that enabled the Internet and the Web. It transcends both models, in a number of ways.

The traditional IT model focused on calculations, data processing, transactions, and other highly structured tasks. It served businesses well for those rigid applications and will continue to do so over time. But the model breaks down when trying to extend it into applications or processes that aren't so highly structured, such as long-term enterprise resource planning projects.

The Internet computing model had a different design point. It gave us simple mechanisms, based on industry standards, to link together many components, which can be used to perform relatively simple functions like browsing and searching for information, and sending and reading e-mail. The Internet computing model enabled a handful of new business models. But more important, it revolutionized the way that existing things were done, especially in the areas of communications between companies, marketing, sales, and customer support.

With that revolution came the recognition that computing technology is exponentially more powerful when it's based on industry standards. That meant the industry would need additional standards and mechanisms to handle more sophisticated applications.

The on demand Operating Environment, as a computing model, builds on both the traditional and Internet computing models, leveraging industry standards to redefine how existing systems and technologies interact. This enables the

creation of a highly modular environment, where application and infrastructure components can be more easily defined and managed. This enables a more flexible and real-time implementation of business policies than was possible with more structured computing models.

We recognize that this isn't a one-size-fits-all solution or methodology. Organizations have different priorities, different personalities. An on demand approach reflects that. With many different entry points, where to get started depends on the organization's priorities and resources.

In today's pragmatic environment, there are only a handful of organizations prepared to tackle all of the facets of creating an on demand business. Most companies opt to start more slowly. They focus on one key process and transform it. Or they start by taking steps to simplify their operating environment, increasing overall flexibility and resilience, while reducing the resources that their current approach requires.

1.2 Infrastructure to support an on demand business

Over the last several years, most enterprises have concentrated on making individual business processes more efficient. This work has typically been done within application or line of business silos. As we look forward, continued improvement in business performance will require a horizontal view, looking across the business and even across the ecosystem of suppliers, partners, and customers.

To create applications and support business processes across lines of business or organizations will require the ability to use and integrate existing applications and processes. This will provide flexibility to allow the business to easily adapt and assemble new applications to support new business requirements. If there was ever an argument for using industry standards, this is it: enabling the business to quickly and seamlessly integrate processes that weren't built to work together, from a variety of vendors. With industry standards, applications don't need to be recreated every time some piece of hardware or software changes or is rewritten to support changes in the dependent processes.

Aside from the business flexibility that comes from the ability to integrate people, processes, and information across the business, the IT infrastructure must also be made simpler and more manageable. This includes support for virtualizing the resources required and automating the management and operations of the IT environment.

The characteristics that enable an on demand Operating Environment are the capabilities that enable business flexibility and simplification of the underlying technology infrastructure.

The first focus is to increase business flexibility through capabilities designed to speed integration initiatives. The ability to connect people, processes and information in a way that allows the organization to become more flexible and responsive to the dynamics of its markets, customers and competitors is critical. It becomes increasingly so as the value net is extended to more tightly integrate partners, suppliers and customers into the business processes.

The second focus is IT simplification, the creation of an infrastructure that's easier to provision, deploy, and manage. This is accomplished through the creation of a single, consolidated, logical view of, and access to, all available resources in a network. Many organizations have become comfortable with the practice of over-provisioning, buying excess capacity so they can handle the occasional spikes that almost every system experiences. Eliminating the practice of over-provisioning by moving to an infrastructure that accommodates dynamic resource provisioning can reduce an organization's capital investments significantly.

To achieve more flexibility and componentization, the infrastructure must evolve from silos of complex, over-provisioned, proprietary hardware and software to an industry-standards-based infrastructure, where capacity can be optimized across the entire organization.

1.3 Capabilities

On demand Operating Environment capabilities enable business flexibility and IT simplification. There are two entry points: integration and infrastructure management. The objective is to evolve to an industry-standards-based, integrated, automated and virtualized IT environment.

Each of the capabilities of an on demand Operating Environment acts as a facilitating element to enable the deployment of an underlying infrastructure that drives business flexibility and IT simplification.

Integration capabilities enable the connection of people, processes, and information in a way that allows businesses to become more flexible to the dynamics of the markets, customers, and competitors around them. To maximize the ability to integrate within and beyond the enterprise, there are six key

capabilities required. These will typically be implemented over time according to the needs of the individual business:

- *Business modeling* enables the graphical depiction and simulation of a business process, including task descriptions, resources required, and decision points.
- *Process transformation* enables existing applications and information to be reused in new ways.
- *Application and information integration* enables multiple information sources and business applications to be combined.
- *Access* extends data and information to new classes of devices and methods of interaction regardless of connection type.
- *Collaboration* allows users to interact in a personalized way with dynamic information, applications, processes, and people.
- *Business process management* can model, deploy, and analyze processes with the goal of managing the end-to-end business process.

Infrastructure management capabilities extend access to, and create a consolidated, logical view of resources across the network. This dramatically simplifies the operating environment, increasing flexibility and delivering broad-based cost savings. Fundamental to this simplification are the concepts of automation and virtualization.

Virtualization is the ability to separate the direct dependency of an application to a physical resource. Through virtualization, an enterprise will:

- Have a single, consolidated view of, and easy access to, all available resources in the network, regardless of location.
- Efficiently access and manage those resources to reduce operations and systems management costs while maintaining needed capacity.
- Respond dynamically to the application needs of its users.
- Gather and access information across the organization quickly to gain competitive advantage.

Automation enables an IT infrastructure to manage many day-to-day tasks itself. With a self-managing infrastructure, efficiency is increased and resource allocation simplified. A fully automated IT infrastructure can sense changing conditions, like surges in demand or isolated application errors, and can spot trends that could lead to costly system downtime. The infrastructure automatically responds by taking corrective actions that ensure IT resources remain aligned with business goals.

To achieve this simplified and optimized management of the infrastructure, the following capabilities are required. Again, these will typically be achieved over time as the business requires.

- *Availability* helps ensure the health and appropriate functioning of IT environments.
- *Security* helps ensure that information assets, confidentiality and data integrity are protected.
- *Optimization* helps make the most productive use of the IT infrastructure.
- *Provisioning* makes the right resources available to the right processes and people at the right time.
- *Policy-based orchestration* senses, triggers, and responds according to business goals.
- *Business service management* helps to visualize the IT environment in business terms and manage service levels to business objectives.
- *Resource virtualization* provides a single, consolidated, logical view of, and easy access to, all available resources in a network (including servers, storage, and distributed systems).

Although we discuss the capabilities through the two entry points of integration and infrastructure management separately, in reality they are tightly linked. Security, for example, permeates IBM solutions, providing a critical, pervasive functionality across the on demand Operating Environment.

1.4 On demand Operating Environment architecture

Figure 1-1 on page 9 represents the on demand Operating Environment architecture.

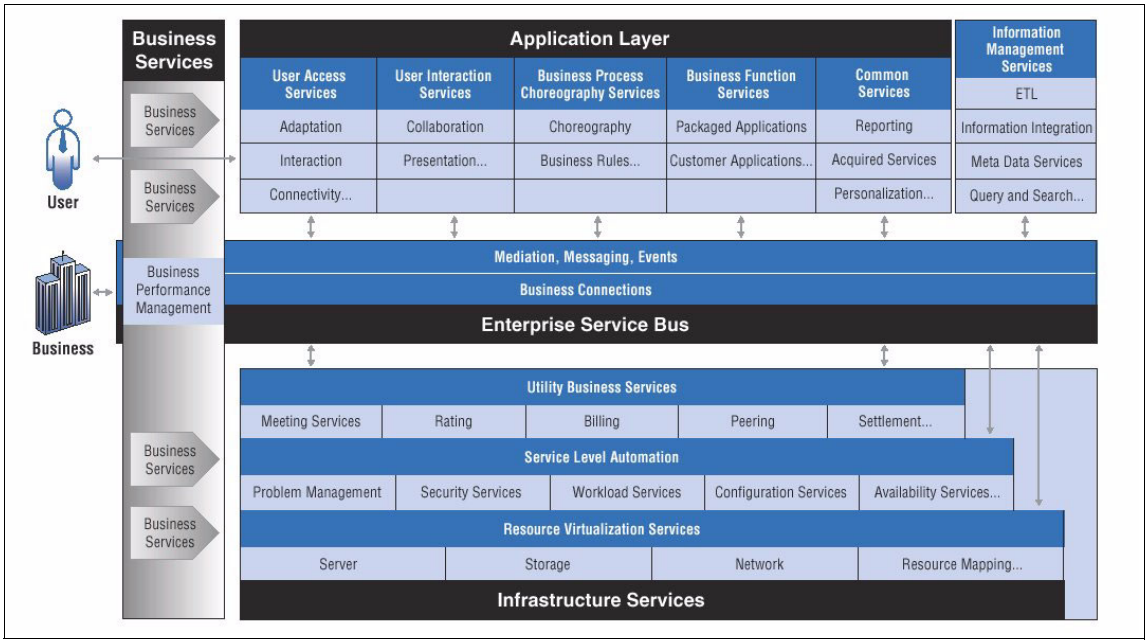


Figure 1-1 Operating Environment architecture

The on demand computing model applies at various levels in the IT stack. At the system level, the components are system objects (for example, computing capacity, storage, and files). At the application level, components are dynamically integrated application modules that constitute sophisticated, yet much more flexible applications. At the business level, the components are business objects, defined for particular vertical industries or more generally, as they apply horizontally across industries. Because the on demand computing model is based on industry standards, it can be used to define the business, applications, and systems at various levels: within a department, across an entire enterprise, or throughout an industry ecosystem. It enables true end-to-end business process integration.

The on demand Operating Environment is based upon the concepts of a Service Oriented Architecture (SOA). A Service Oriented Architecture views every application or resource as a service implementing a specific, identifiable set of (business) functions. Services communicate with each other by exchanging structured information—messages or documents (sometimes called business objects). Their capabilities are defined by interfaces declaring messages they can produce or consume, policy annotations declaring quality of service required or provided and choreography annotations declaring behavioral constraints that must be respected in service interactions. The actual implementation is hidden from the requester of a service, thus Service Oriented Architectures are a

convenient way to achieve application integration by allowing new and existing applications to be quickly combined into new contexts. Existing applications are “adapted” to service declarations. The interaction of services can be direct, or can be mediated through an intelligent infrastructure, which we will call an 'Enterprise Service Bus (ESB).

Service Oriented Architectures require standards for the definition of services and their capabilities and interactions. The adoption of this architectural approach has been greatly facilitated by the growing acceptance of XML use to provide a standard representation of structured information and of “Web Services” standards (often called WS-* standards). The conceptual model of a Service Oriented Architecture applies to the virtualization of both business functions and physical infrastructure. It spans the construction of applications as well as their deployment and management. A client (user or business) only sees a collection of business services, and is interested in their quality of service, but is shielded from the details of application assembly and service delivery through the on demand Operating Environment.

1.5 Summary

In this chapter we have introduced the on demand Operating Environment and briefly described its key capabilities, characteristics, and architecture. An on demand Operating Environment is not a specific product or suite of products, and it is not something that will be created or deployed overnight. Enterprises will evolve to the Operating Environment by deploying various capabilities based on the specific needs of their business.

An on demand Operating Environment provides businesses with flexibility by enabling integration between people, processes, and information, and creating a manageable infrastructure through automation and virtualization.

In the next chapter, we describe the Infrastructure Management capabilities in more detail and then in the second part of the book we describe how to get started today by addressing some common business requirements that can be met through an on demand Operating Environment.

2

Integration overview

This chapter discusses the role of integration in creating, deploying, and managing an on demand Operating Environment.

It provides a high-level overview of the drivers for integration and describes a framework with which to view the integration functions. It looks at the key technology components of this framework, and also discusses some of the non-technical factors that are critical to the success of any integration effort.

The framework provides for a step-by-step implementation, so that enterprises do not have to implement all components to achieve business value. This chapter considers several approaches that customers may consider for getting started, based on their current “pain points” and priorities. Additional capabilities can be added as required based on business objectives.

2.1 Business drivers

When evaluating how IT can support the business requirements of an enterprise, several questions are often asked by the CIO:

- Can I react quickly enough to opportunities or threats in my environment?
- Can I create new business value from my existing IT systems?
- Can users react in real time to the most recent information?
- Can I easily access information anytime, anywhere, with my choice of devices?
- Are my business operations integrated end-to-end for optimal efficiency?

In many of today's organizations, business success is often tied to the speed with which business strategies can be changed to counter competitive pressures and capitalize on opportunities. In other words, rapid time to value. At the core of e-business on demand is business integration, the combining of resources (people, process, and information) to optimize operations within an enterprise and with its partners and customers. Integration maps directly to the business requirement to be more flexible.

Transformation into an on demand business requires building a dynamic infrastructure based on tightly integrated, streamlined critical business processes, processes that are efficiently linked across a company and with those of key trading partners, suppliers, and customers. Integrated business processes provide flexibility and the ability to respond immediately to almost any customer demand, market opportunity, or external threat.

The true integration challenge is responding to these business demands quickly, by overcoming the integration requirements of the highly complex, underlying silo-oriented business systems that have evolved over the years.

To gain this flexibility, a well thought out integration strategy based on a robust infrastructure is key. This infrastructure provides automation and management of value chain processes in the extended enterprise, internally as well as with partners and customers. This capability is key to reduced “time to value,” that is, reduced cycle times and costs, and business agility in the face of competitive pressures.

Key attributes and abilities that are required, include:

- People
  - Standardized access to applications
  - Access any time, any place

  - Dynamically adaptive role-based workplaces
- Process
  - Model processes
  - Integrate applications
  - Connect externally
  - Monitor processes
  - Manage business results
- Information
  - Leverage data and content resources
  - Access data in place
  - Consolidate data
  - Transform data
  - Manage data placement

2.2 Framework for integration

Providing business flexibility effectively in any large, complex enterprise requires an interplay and collaboration between several organizational functions, roles, and technologies. Therefore, any framework that addresses this significant opportunity and challenge in the real world must include not only the technology component of the equation, but also the methodology and governance model by which the technology and tools will be applied, to increase the probability of success. This section provides a brief discussion of these elements.

The primary components of an integration framework include:

- Access and collaboration
- Business process execution
- Enterprise Service Bus
- Adapters
- B2B connectors
- Common information and resource model

These components are described in more detail in the following sections.

2.2.1 Access and collaboration

As information flows through the business process, many individuals need to interact with that information in different ways. For instance, a business analyst may extract relevant data for a step along the process. A reviewer verifies the information and identifies data gaps to query if necessary. The approver signs off on the particular process step. Many other people interact and collaborate along the way. These collaborative and ad-hoc workflow capabilities are key to providing flexibility to the people in an on demand business as they complete the tasks in a business process.

The collaborative and ad-hoc workflow capabilities discussed here are different from those of a more structured workflow process, such as an insurance claims process or a mortgage approval process. However, within individual tasks or exceptions of such processes, the ability for individuals and teams to interact greatly enhances the overall effectiveness and efficiency of the entire business process. Collaborative tools tie all elements of collaboration directly to the business processes. This allows the content from collaborations taking place throughout the process to be associated and transmitted with a business process, providing better context, which assists in decision making by enhancing the accuracy of information flow and cutting down on the amount of misinterpretation of information. The collaborative tools can be in various forms including teamrooms, instant messaging, Web conferencing, or simple e-mail. An important characteristic of these tools is that they allow rapid set-up and take-down of the environment promoting collaboration. This reflects the real-life dynamics of ad-hoc teaming, to address a particular issue, solve a problem rapidly, or bring various domain experts together for a project with a defined duration, after which that particular team is no longer involved in addressing the same issue.

The collaborative capabilities discussed can be provided as “modular” services that can be aggregated quickly to address a business process issue. To integrate these collaborative services, an environment that encapsulates them in standards-based services is required. This environment should provide the following end-user services:

- Aggregation of content and services for the individual user.
- Personalization of content based on the profile of individual users.
- Dynamic interaction between the various services presented to the end user.
- Authentication of the end-user and passing of these credentials to other services to determine access privileges.

The capabilities described here are provided through a “portal” framework. The portal framework provides a standards-based mechanism for aggregating and viewing information. It is easy to use, enables personalization of content based

on a user profile, and allows for dynamic interaction between various services presented to the end-user through a construct called a “portlet.”

A portlet (similar to a servlet) provides access to a specific application or function being made available to the user via the portal. Toolkits are available to enable developers to customize and manage the enterprise portal, and to create, test, debug, and deploy individual portlets and Web content. Templates enable developers to quickly and easily create their own portlets. Debugging and deployment tools shorten the development cycle.

Portals also allow for common services such as single sign-on for Web applications.

The collaborative tools and portal infrastructure described here greatly enhance the productivity and flexibility of individuals and teams working on a business process. The tools also reduce the probability of errors by ensuring a consistent flow of information between people working on a task or across tasks. These capabilities often provide a critical edge to innovative on demand businesses and provide rapid time to value.

Further details on access and collaboration are discussed in Chapter 5, “How to react in real time through seamless flow of information” on page 139.

2.2.2 Business process execution

Business process execution defines how various processes involving people and applications interact together and with various other resources to effectively and efficiently complete a business process. This is sometimes referred to as “choreography.”

This choreography represents the collaborative effort of various functions in an organization to rapidly model business processes, deploy them to a run-time IT infrastructure efficiently, monitor them and measure their performance based on business metrics.

In many cases, organizational linkages between the business analysts in the line-of-business organization and other functions, such as the IT organization, may not be strong. Therefore, the flow of information between the various organizational units and functional roles is sometimes cumbersome, with very few tools to facilitate it and help leverage the work that has already been done by others.

Typically, IT organizations in customer organizations have been early adopters of technology-based tools to model, design, deploy and monitor an IT environment. The adoption of tools by business analysts for business process design and modelling has been slower. As a result, the correlation of systems to

business value, and metrics such as return on investment and time to value have been weak.

Linking the business process analysis function closely with the IT function allows an organization to more effectively and rapidly adapt to changes in the business environment. It also facilitates the measurement of IT investments with business metrics closely correlated to an enterprise's business objectives, rather than to purely IT-based metrics.

Business process execution provides the development, deployment, and run-time tools to facilitate rapid redesign of business processes and the resources associated with them. It also provides the tools to link the various roles and functions in an organization that are involved in this collaborative effort, as shown in Figure 2-1.

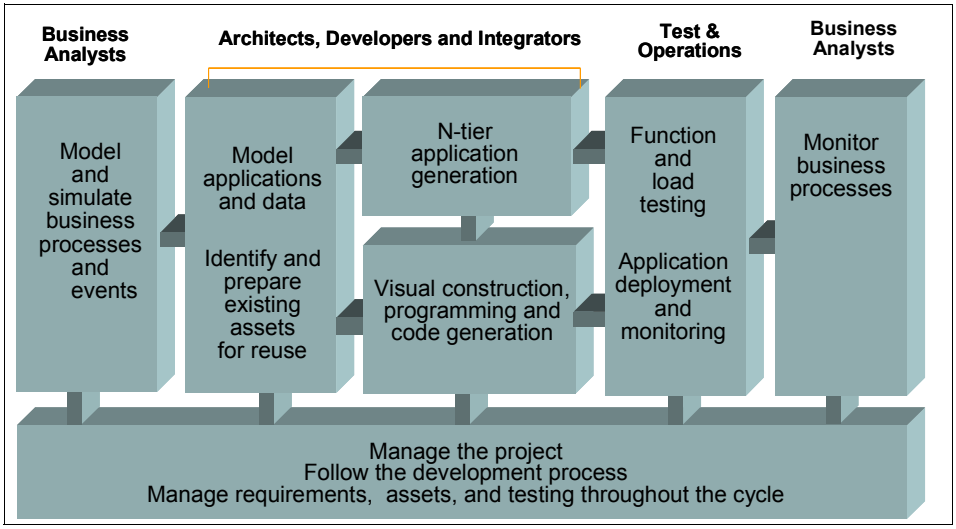


Figure 2-1 Phases in business process execution

One of the key factors in enabling an end-to-end description of workflow, from business process analyst to architect to implementer, is the availability of standards that can describe the workflow and provide linkage between the tools. Standards for describing workflow and linking the various toolsets, such as BPEL, are in a nascent and emerging stage. IBM is working actively with its business partners and vendors to define and establish these standards so that the tools for the various functions can be more closely linked.

The vision for an integrated toolset is shown in Figure 2-2. It allows multiple entry points based on the skill set of a particular organization and its governance model. The toolsets are integrated and linked based on standards for workflow

such as BPEL, the open standards environment Eclipse, and others. The systems developed through the consistent use of such toolsets are envisioned to be deployed in an open standards, J2EE-based run-time workflow environment that supports Web Services for maximum flexibility.

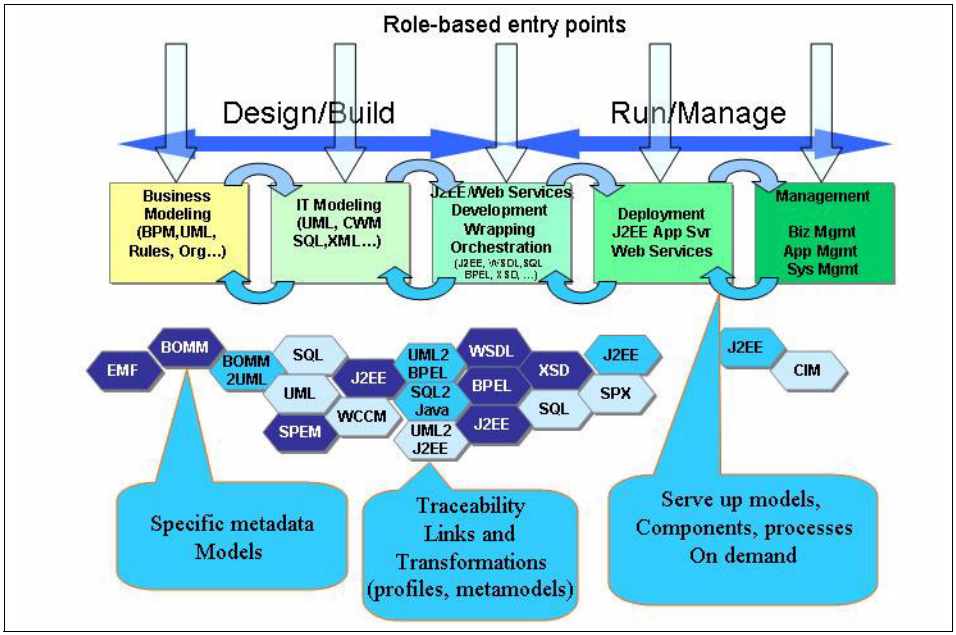


Figure 2-2 The vision for model-driven business integration

2.2.3 Enterprise Service Bus

Given today's business climate and requirements, business success (and at times, even survival) requires broad interoperability:

- Intra-enterprise
- Inter-enterprise
- Across heterogeneous sets of applications, platforms, and languages

The on demand infrastructure must provide a level of integration and federation across the heterogeneous and distributed computing environment that, by its very nature, consists of various platform architectures, programming languages, access protocols, and implementation technologies.

This level of integration is provided by a set of enabling technologies and architectures around it such as the Services Oriented Architecture (SOA). The SOA views all functions in a system as requestors or providers of IT "services."

“Web Services” is an enabling technology that provides an instance of the SOA, and is discussed later in this section.

A key component of the SOA is the Enterprise Service Bus (ESB). The ESB can be thought of as a pre-packaged SOA implementation containing the necessary functional components to achieve the aims of an SOA. It provides a mechanism to connect the various applications and information sources in an enterprise in a manner that can be mapped to the business process being considered. It is based on open standards and allows effective connectivity of applications both within an enterprise, and outside the enterprise to partner systems.

Service Oriented Architecture

To reach the key business objectives of flexibility and rapid time to value, companies need loosely coupled business processes. These business processes can address current inefficiencies in applications and processes that tie a company to its customers, trading partners, or suppliers. They can also help it better compete for customers through integrated collaboration.

These loosely coupled business processes can be built quickly and easily based on a framework known as a Service Oriented Architecture (SOA), which envisions a system as a collection of modular services that can be aggregated and mapped to actual business processes.

There are three major roles that can be identified in the SOA as shown in Figure 2-3.

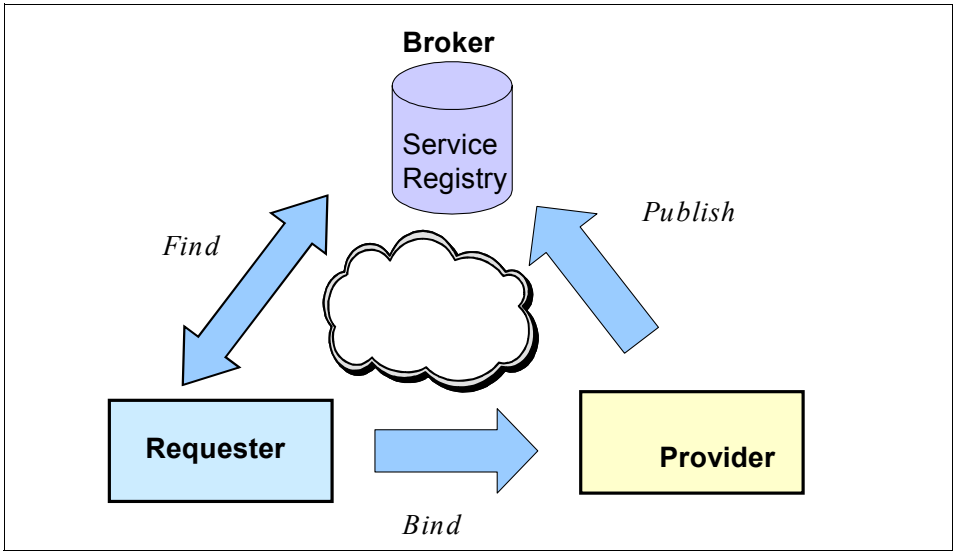


Figure 2-3 Roles in the Service Oriented Architecture

- The *service provider* creates a Web Service and publishes its interface and access information to the service registry.
- The *server broker* (also known as *service registry*) is responsible for making the Web Service interface and implementation access information available to any potential service requestor.
- The *service requestor* locates entries in the broker registry using various find operations and then binds to the service provider to invoke one of its Web Services.

Through SOA, business applications can be easily published and discovered by businesses because they are described in a uniform way on a network, based on Internet protocols. The entire “Publish-Find-Bind” process allows e-business applications to be connected more easily both inside and outside the enterprise.

Business processes are best implemented with services, particularly with a service oriented architecture that supports their development as well as their deployment. With a service oriented architecture, functions within applications are defined as services. Business processes therefore are a collection of services—inside and outside the enterprise—connected as needed through a standard interface. Applications that use services need not be aware of the service details at the time the application is developed.

Services and a service oriented architecture offer a giant step in reducing the complexity as well as the costs and risks of new application development and deployment because they provide:

- Reduction in intra-enterprise and inter-enterprise development and deployment dependencies.
- Reduction in the amount of training developers have to undertake. Developers merely need to understand the interface to a particular service, instead of an entire system.
- Reduction in the size of projects. Developers create components or services one at a time, instead of working on an entire system.

Web Services

Web Services is an enabling technology that provides an instance of SOA. There are many definitions of Web Services:

- A Web Service is a collection of functions that are packaged as a single entity and published to the network for use by other applications. Web Services are building blocks utilizing the component object model for creating open distributed systems, and allow companies and individuals to quickly and cheaply make their digital assets available worldwide.

This definition is somewhat loose, but illustrates that Web Services are a means to expose one's enterprise assets to a wide range of applications.

- Web Services are a new breed of Web application. They are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web Services perform functions, which can be anything from simple requests to complicated business processes. Once a Web Service is deployed, other applications (and other Web Services) can discover and invoke the deployed service.

  This is a more formal definition and is the one used by IBM.

- A Web Service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described, and discovered by XML artifacts, and that supports direct interactions with other software applications using XML-based messages via Internet-based protocols.

  This is the working definition of a Web Service that the W3C Web Services Architecture group has agreed upon.

Based on these definitions, a Web Service has to support the following:

- Open XML description of the service's form and function to enable the service to be published, located, and invoked/bound.
- XML-based messages between different software applications.
- Open Transport Protocols such as HTTP, SMTP, TCP/IP, and so on.

Web Services is an instance of the service oriented architecture that enables the provisioning and the consuming of services (Web Services), providing tools and infrastructure to the different organizations' operators for specifying, publishing, finding, and binding services.

Web Services are based on a layered functional framework. They encompass various protocols for service discovery, publication, description, messaging, and transport.

Benefits of Web Services:

- **Web Services promote interoperability:** The interaction between a service provider and a service requester is designed to be completely platform and language independent. This interaction requires a definition of the interface and service independent of the implementation platform. This is done using a Web Services Description Language (WSDL) document to define the interface and describe the service, along with a network protocol (usually HTTP). Since these descriptions are independent of the implementation platform, interoperability is enhanced.

- **Web Services enable just-in-time integration:** While service requesters use service brokers to find service providers, the discovery takes place dynamically. Once the requester and provider have found each other, the provider's WSDL document is used to bind the requester and the service together. This means that requesters, providers, and brokers work together to create systems that are self-configuring, adaptable, and robust.
- **Web Services reduce complexity through encapsulation:** Service requesters and providers concern themselves with the interfaces necessary to interact with each other. As a result, a service requester has no idea how a service provider implements its service, and a service provider has no idea how a service requester uses it service. Those details are encapsulated inside the requesters and providers.
- **Web Services give new life to legacy applications:** An application can be "wrapped" in a a Simple Object Access Protocol (SOAP) wrapper and generate a WSDL document to cast it as a Web Service. SOAP is the industry standard protocol for carrying XML messages and is a key Web Services technology. This allows legacy applications to be used in interesting new ways. In addition, the infrastructure associated with legacy applications (security, directory services, transactions, and so on) can also be "wrapped" as a set of services.

Considerations with Web Services

Some Web Service issues to consider are:

- Binding to Web Services dynamically requires that the contents of the UDDI registry be trusted. Currently, only private UDDI networks can provide such control over the contents.
- The SOAP server footprint is significant and the technology is relatively new, so adding the Web Service provider stack to existing enterprise systems can be a consideration.
- Standards for Web Services are still evolving. Of particular note here are standards for transactional Web Services and standards for Web Services security. IBM is actively involved, along with other industry members (such as Microsoft®), to develop and gain approval of these standards. However, this work is still evolving.

Enterprise Service Bus

The intense interest in IT-supported business integration has resulted in various approaches to connecting the diverse applications and resources within an enterprise and with its partners. A key driver has been providing business and IT flexibility to adapt to rapidly changing conditions.

The Enterprise Service Bus (ESB), which can be thought of as a pre-packaged SOA, provides a mechanism to connect the various applications and information sources in an enterprise in a manner that can be mapped to the business process being modelled. Based on open standards, it allows effective connectivity of applications both within an enterprise and outside an enterprise to partner systems.

Figure 2-4 is a high-level depiction of an ESB.

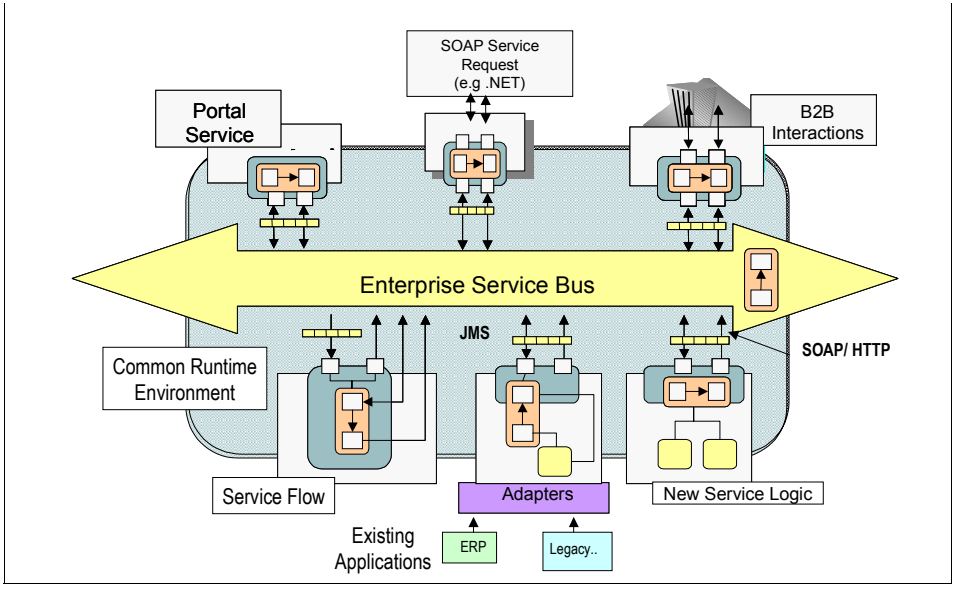


Figure 2-4 Enterprise Service Bus

The ESB provides a standardized bridge between the producers of services and the consuming applications in a system. The ESB provides a logical view of sets of services that are available for use. These are invoked by a common interface and have a common management architecture. The ESB also provides common services required by other services such as access control and discovery of other services.

As discussed previously, decoupling the implementation of a service from its interface allows a great deal of flexibility in building systems that can be adapted and changed rapidly to reflect business processes.

A key enabling technology used in implementation of ESBs are Web Services. However, ESBs need to integrate Web Services-enabled applications and applications that have not been Web Services enabled to ensure that customers can leverage their investment in existing systems.

Therefore, an effective ESB solution must allow for:

- Decoupling of the interface of services from their implementation on a particular system to provide flexibility and ensure that integration is not technology-specific.
- Intelligent routing to choreograph and map the services provided on the ESB to actual business processes.
- Multiple types of standards-based inter-service communication approaches such as asynchronous message-based ones (Java™ Messaging Service - JMS), as well as publish and subscribe, to provide the most flexibility in mapping to business processes.
- The ability to connect applications that have not been Web Service enabled through other standards-based technologies such as JCA connectors and adapters. This allows customers to take an evolutionary approach to integration and leverage their existing investments.
- Transformation of information between various formats expected by the interacting services. To the extent that XML formats and services are leveraged in the ESB, information sent between services could be self defining.
- Common system services to provide security, access control, manageability, and quality of service for the services interacting on the bus.
- Scalability based on a service bus architecture rather than hub and spoke.

An implementation of an ESB will generally contain the following components, relying heavily on Web Services technology to provide its functionality:

- **Web Service-compatible integration server**: An application wishing to invoke a component on the network or expose its services on the network will do so via an integration server designed for the purpose. Quality of service aspects relating to the service's exposure on the network will be declared by the application, and managed transparently by the integration server.
- **Standardized interfaces between applications and the integration server**: An application created using a given language can invoke the services of the integration server. This service is requested using a standard API linked to the language used (for example, Java, C# or Perl), which isolates the application from any proprietary integration server API.
- **Standard Web Service communication protocol**: Natively integrates the notions of quality of service that are essential for wide-scale application integration (confidentiality, integrity, non-repudiation, guaranteed delivery, transaction, sessions, administration, and so on). This 100% XML protocol makes it possible to progressively integrate other integration servers from other vendors into the architecture, and to integrate a network of compatible servers from a partner without harming quality of service.

- **A container for Web Services and orchestration**: This is incorporated in the integration server mentioned previously to support various components relating to the integration logic (message transformation, enrichment, management of long processes, and so on). These components are additional standards for Web Services, which isolates them from the proprietary aspects of the container.
- **A standard administration protocol**: This enables a third-party administration console to supervise a Web Service-compatible server or a network of Web Service servers, thanks to a standardized XML administration protocol.

According to a Gartner report, “An Enterprise Service Bus (ESB) is a new architecture that exploits Web Services, messaging middleware, intelligent routing, and transformation. ESBs act as a lightweight, ubiquitous integration backbone through which software services and application components flow.” (Roy Schulte, 12/9/2002)

This definition is very similar to that of “Enterprise Application Integration” (EAI) tools, apart from three significant areas: Web Services, ubiquity and weight. Enterprises are just starting to measure the impact of application integration. Some of the issues with traditional EAI that customers have wrestled with are:

- Proprietary development interfaces
- Proprietary communication protocols
- Lack of consistency between application servers and EAI solutions
- Licensing costs
- Relatively low scalability

In addition to EAI, other approaches to integrate applications (such as DCE and CORBA) have met with limited success. Because of its standards-based approach, it appears that ESBs may be able to overcome some of the issues just mentioned. Additional factors that may contribute to the probability of success are:

- Application integration today is more closely tied to business process re-engineering and strategic business objectives.
- By leveraging Web Services, ESBs are taking a bottom-up approach that allows customers to start small and build on their experience and investment.
- ESBs support XML and industry standard (W3C) schemas.

An area of concern for customers is that the standards for Web Services and other enabling technology building blocks in the ESB are still emerging, and will take some time to achieve mainstream adoption.

In particular, still missing are the vital standards covering the guarantee of asynchronous message delivery and security. The HTTPS standard does not provide high enough security for Web Services and the RPC connotation of SOAP/HTTP contradicts the notion of loose coupling on which Web Services are based.

2.2.4 Adapters

A key element of any integration effort is to connect the various systems and activities that comprise daily operations. Not all of the systems that an enterprise has invested in are or can be enabled for Web Services.

Adapters facilitate the exchange of data and transactions and provide a mechanism to integrate these “legacy” and package applications into the ESB discussed in the previous section.

Adapters are generally used to integrate applications into an application server or an Enterprise Integration Server (EIS), which can then “wrapper” these applications into the appropriate services to be integrated via the ESB. Several ISVs and package software vendors provide adapters to facilitate the integration of their systems. This is shown in Figure 2-5.

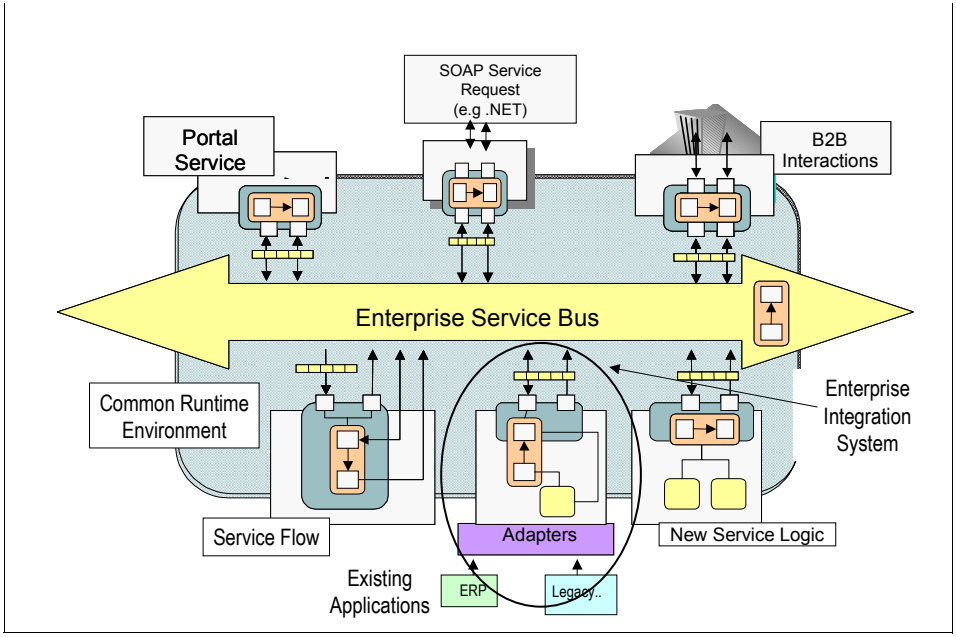


Figure 2-5 Adapters and Service Oriented Architecture

The decision to use an application server or EIS system will depend on the number of applications being integrated and the differences in data format expected by each system. The connection complexity between systems increases exponentially with the number of systems integrated. The situation could be further complicated by the differences in data format expected by each system. Therefore, in situations where there are a large number of diverse systems to be connected, it may be of value to use an EIS system with adapters to simplify connections and transformation. Several EIS systems also have pre-defined process templates that can be used to reduce the time and risk of an EIS project.

Adapters are generally based on standard architectures such as J2EE. The adapters based on J2EE standard (JCA) provide common services to the connected application, including the programming interface, system services such as connection pooling, transaction and security management.

2.2.5 B2B connections

In the current business environment, an enterprise needs close linkages with its business partners in the value chain. Visibility into their operations and integration with the business processes of the enterprise are essential to enable rapid reaction to changing market conditions and rapid time to value for new products and services.

As discussed earlier in this chapter, the concept of an ESB can be extended beyond the enterprise to achieve business flexibility, with the appropriate security and access management controls. Figure 2-6 shows an ESB used to support business processes within and beyond an enterprise.

B2B connections are the components of the integration framework that allow mapping of services outside the enterprise onto the ESB to create a more seamless extended enterprise. This allows the business process choreography concepts discussed earlier to be used in the extended enterprise as well, for greater flexibility and efficiency.

Several options may be explored depending on the nature and integration requirements of the external partner systems. The partner may have a Web Service-based system, in which case a loose coupling onto the ESB may be a relatively simple concept to implement. Alternatively, the partner system may use a more traditional integration interface such as EDI or messaging. In such situations the business connection component will need to map the information from the partner system into a service that can be deployed and consumed by a business process via the ESB.

In either case, there are a couple of considerations to keep in mind over and above the integration of intra-enterprise systems.

- The systems interacting across enterprise boundaries must be more de-coupled than those in an intra-enterprise environment, since there is no unified control point. If one of the partner systems expects a synchronous type of communication, there are messaging-based implementations that present a synchronous interface to the partner application. This “de-synchronized” link effectively acts as a proxy or buffer between the asynchronous and synchronous communications. One possible example of implementing this is by a “one-way” invocation technique as shown in Figure 2-6.

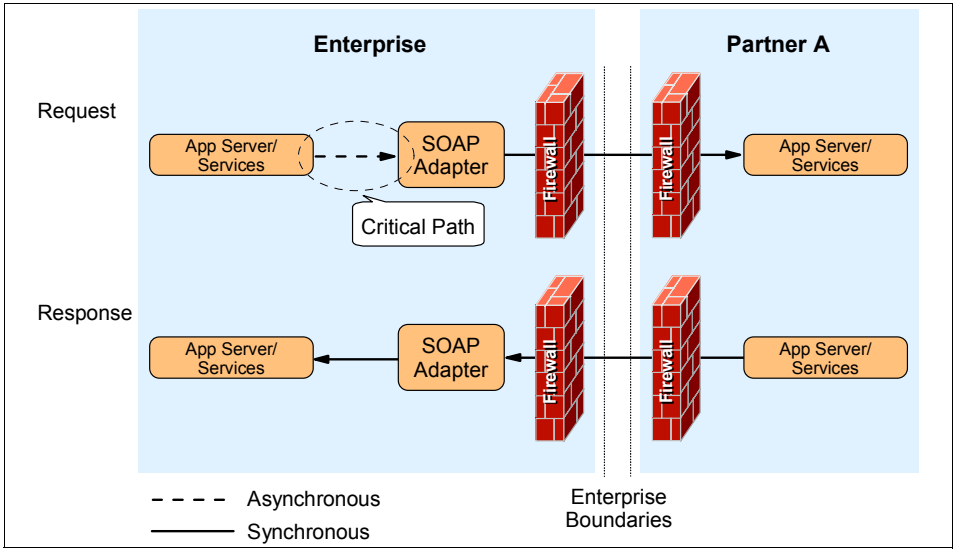


Figure 2-6 Decoupling intra-enterprise integration with one-way invocation

The advantages of this approach are:

- The source application is decoupled from the behavior of the channel between the enterprises.
- The source application is decoupled from the behavior of the target application in the partner enterprise.
- The source application is decoupled from any intermediary (such as a gateway) which is used as an exit point from the enterprise.

Some disadvantages of one-way invocation are:

- Services provided by target applications must only implement a one-way transmission style.

– The delivery of the request message is not reliable.

- In many situations the partner system may not directly support Web Services-based integration. An example is EDI systems, which have been widely deployed for some time, and which generally use proprietary interfaces and yet provide significant value in terms of partner integration. However, customers are looking for ways to reduce costs without replacing their applications, by using standards-based Internet technologies for exchanging EDI messages.

  There are other legacy systems that also do not have a Web Services-based interface, but provide significant value in an inter-enterprise situation.

  One of the alternatives that can be considered to connect partner systems to the enterprise ESB in such a situation is the use of a connector, as shown in Figure 2-7.

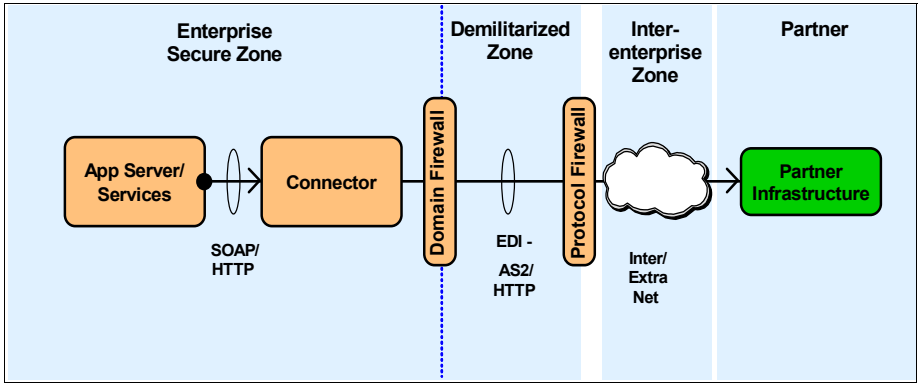


Figure 2-7 Inter-enterprise connection

Product implementations using WebSphere® for this type of approach are discussed in Chapter 4, “How to rapidly modify business processes” on page 63.

- There is an additional level of security and access control to consider since vital information in the organization may be exposed to external processes. This added level of security and audit trails may be especially necessary in some environments due to regulations governing the privacy and access of information. An example is the Health Information Portability and Accountability Act (HIPAA) in the US, which provides regulatory guidelines for privacy and security of medical information with potential criminal penalties for intentionally violating them. The higher the value of a service an enterprise offers, the more levels of security and access control may be necessary in an extended enterprise environment. Figure 2-8 illustrates, conceptually, a multiple security level implementation of partner communications.

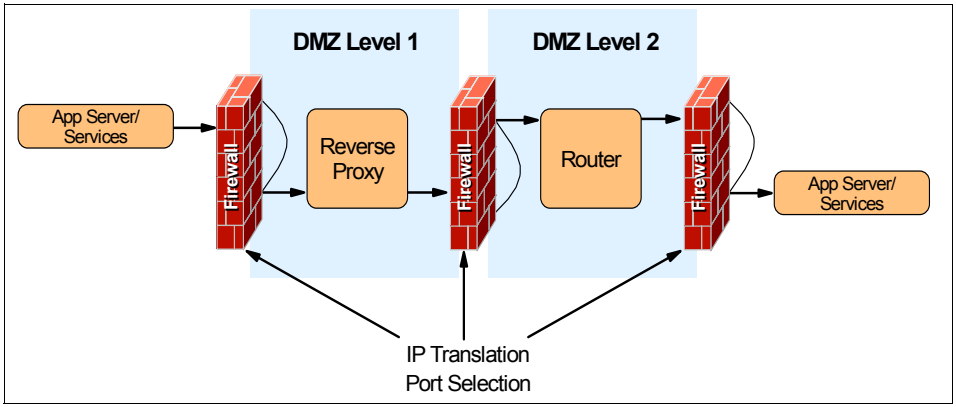


Figure 2-8 Inter-enterprise security with a dual level DMZ

Additional details and considerations for extending integration methodologies beyond the enterprise are contained in the redbook, *Patterns: Direct Connections for Intra- and Inter-enterprise,* SG24-6933.

2.2.6 Common information and resource model

The common information and resource model provides a consistent view of information from all the diverse sources within an extended enterprise, including internal information as well as that from partners. On demand businesses must be able to find and realize the value of this information, in real time, independent of how it is stored.

Several things must happen to ensure the seamless flow of information:

- Enterprises must make the information easy to interpret. This requires the ability to normalize fields such as date, time, name, and more so that they can all be compared and viewed in the same way. With normalized data fields, analytics can be applied to that information to assist with more complex interpretation, like identifying patterns in the data.
- Enterprises must make information in all its forms accessible to employees, customers, partners, and suppliers in a way that makes it simple to interpret and take action on in real time.
- Users require the flexibility to define what kind of information they are interested in, independent of where the “real” information is stored or what application generates and maintains it.

Today, common application domains such as customer relationship management (CRM), supply chain management (SCM), and enterprise resource planning (ERP) often seek to present a single, unified view of logical business

entities to their users. Such entities might include customers, accounts, orders, and so on. Unfortunately, critical information appropriate to these business objects is often dispersed through different systems. Someone, or some service, must locate the necessary information, access it using often proprietary interfaces, resolve any discrepancies as needed, and consolidate the data into the desired form.

Locating the information can require modest or substantial effort, depending on how (and where) it is managed and what degree of documentation is available about its characteristics. Retrieving (or modifying) the information almost always poses a problem; syntactic, semantic, and performance issues must be taken into consideration to prevent run-time errors, ensure efficient use of system resources, and achieve acceptable levels of response time and throughput. However, even after the necessary remote information is successfully located and accessed, it often must be translated, transformed, and cleansed before being suitable for broader use. Again, such work may require substantial programming, as well as considerable knowledge about the source and target problem domains. Finally, merging, correlating, and consolidating the data effectively requires sophisticated design and implementation skills.

Information integration enables companies to create new views of enterprise information. The goal of information integration (II) is to enable the following:

- **Federating and optimizing data access**

  This is the ability to transparently access diverse business information, from a variety of sources and platforms, as though it were a single resource. This capability enables enterprises to rapidly construct virtual databases tailored to specific application requirements. This virtual repository is an intelligent facade over a collection of information sources that enables queries over both individual sources and combinations of sources to be efficiently executed. IBM DB2® II federation supports real-time, ad-hoc query using an abstract relational view across diverse data, uses existing reporting and development tools, and relies on leading-edge cost-based optimization. A federated server may access data directly (by accessing a relational database) or access an application that creates and returns data dynamically (such as a Web Service).

  A key question that comes up is how to position data federation versus data consolidation, both of which are similar concepts and deal with requesting and receiving data that lies outside the immediate database. *Data federation* operates in real time, generally on operational data stores, while *data consolidation* generally sets up data in advance of queries through the Extract, Transform and Load (ETL) process. Data consolidation requires more storage, while data federation could impact the performance of operational data stores due to its complex queries.

High-level advantages of data federation over data consolidation include:

- Real-time or near real-time access to rapidly changing data is possible.
- Direct and immediate write access to original data is possible.
- The technical complexity of using data copies is avoided.
- Costs are lower compared to data copies and access.
- Legal issues related to copying of data are avoided.
- Ad-hoc needs of users are supported

Some of the disadvantages of data federation over data consolidation include:

- Access to historical or trend data is hard to obtain.
- Data access performance and availability may be an issue in complex queries.
- It is less applicable to complex transformations, such as joins that are long running.
- It does not perform as well as data consolidation when user needs are predictable and repeatable.

In many cases a combination of data federation and data consolidation may be used, such as with Materialized Query Tables (MQT), which are pre-processed data from remote data sources stored at the local federated server.

- **Incorporating heterogeneous data via XML**

XML has gained wide acceptance as a standard for data interchange. The number of applications that utilize XML as the data interchange format are growing at a rapid pace.

Although XML solves many problems by providing a standard format for data interchange, some challenges remain. When building an enterprise data application, questions such as the following must be addressed:

- What kind of information must be shared between applications?
- How can I quickly search for the information I need?
- How can I have a particular action, such as a new entry being added, trigger an automatic data interchange between all my applications?

These kinds of issues can be addressed by a database management system. By incorporating the XML information and meta-information directly in the database, XML data becomes more accessible for integration and federation as a database object.

- **Sharing information across the extended enterprise and providing a robust infrastructure for business and application integration through Web Services**

  Storing the Web Service description in an appropriate repository offers the potential for interested parties to discover its existence, potentially generating new business for the Web Services provider.

- **Replicating data and managing copies of data**

  Replication is the process used to manage and propagate data changes between multiple copies of data.

  Replication involves the automated and reliable movement of business information from one environment to another for on demand access at target locations. This may involve replicating subsets of data from one system to many (for instance, synchronizing pricing information in retail stores with corporate headquarters), or replicating selected data from many sources to several distributed organizational units (for instance, consolidating information for reference or analysis). The replication server enables management of data movement strategies and monitoring of the synchronization processes.

- **Information caching**

  Information caching combines federation and movement techniques by allowing administrators to create partial local copies (Materialized Query Tables) of remote information that are maintained by replication. Applications can dynamically specify when they can tolerate potentially stale (cached) data. If enabled by an application, queries may be transparently fulfilled from the cache. If the application needs guaranteed access to the most recent data then it can be routed to the remote source.

- **Data mining, content aggregation, federated search and analytics**

  This aspect of information integration provides a comprehensive enterprise content management infrastructure. Large collections of digital content such as text, graphics, images, video, Web content, and more, can be managed and integrated with leading applications such as customer service, ERP, and asset management.

  Information integration provides intelligent mining so users can gain new business insights and harvest valuable business intelligence from their enterprise data.

Information and application integration

It is important to position the various views of business integration. Figure 2-9 provides a high-level representation of various views.

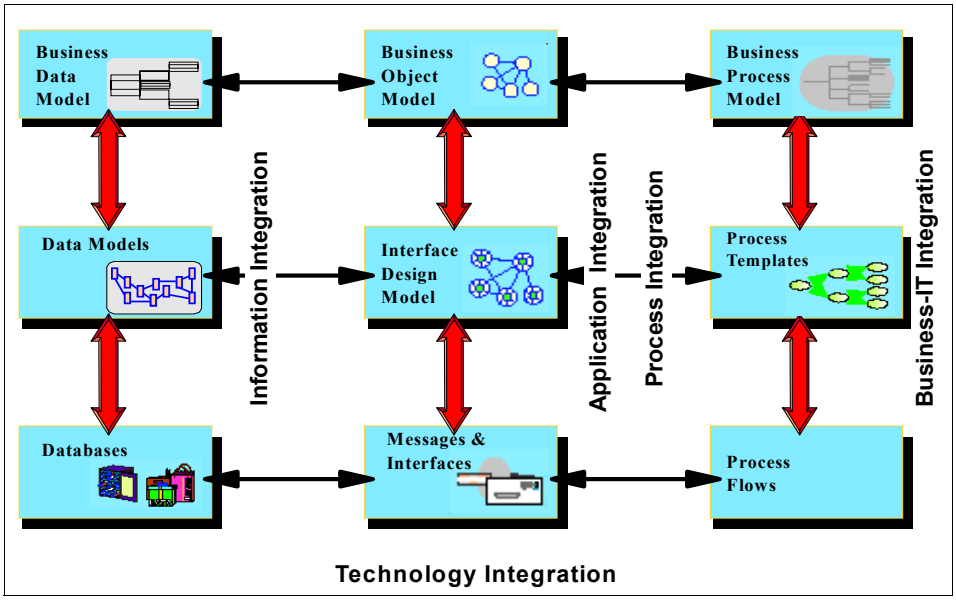


Figure 2-9 Views of business integration

As Figure 2-9 shows, it is particularly important to position information integration and application integration. Application integration is used to communicate business events or single events. Information integration is used to access the state of the business as represented by the information recorded in its data stores.

Information integration focuses on integrating data, while application integration focuses on integrating application programs. Historically, applications were designed with the data schema tightly integrated with the application code; hence blurring the boundary between data and applications.

New technologies, specialized organizations for managing the two types of assets, and integration enablement tools have focused on separating these two closely related assets to maximize their respective and unique values to the business. Therefore, it makes sense to integrate at the data level in some cases and at the application level in others.

Furthermore, most modern integration needs are for a complex mix of both application and information integration. Solution providers have multiple ways to integrate, share, and distribute information. The key is to match the benefits of the different approaches to the different business integration requirements.

Information integration provides the following benefits:

- Data and content may be integrated without moving the data or changing the platform.
- Diverse and distributed data can be accessed as though it were in a single “virtual” database.
- Built-in services allow substantial reduction in design and implementation resources and time.

On the other hand, information integration has the following considerations relative to traditional application integration:

- It requires that the data model is exposed and available. This may not be the case in many legacy and packaged applications. In some cases, even if the data model is available, licensing requirements do not allow direct access to the data. This requirement may not be an issue in some situations where a messaging-based interface can be used to access business logic through DB2II.
- Information integration is generally used for query-oriented, decision support applications. It is not the primary choice in heavy transaction processing environments involving multi-phase commits and so on.

The advent of Web Services as a standards-based technology has provided a foundation for closer convergence of informational integration and transactional application services. This greatly enhances the flexibility and cycle time for a business in getting an aggregate view of the information in the extended enterprise including business partners. Conceptually these capabilities are shown in Figure 2-10.

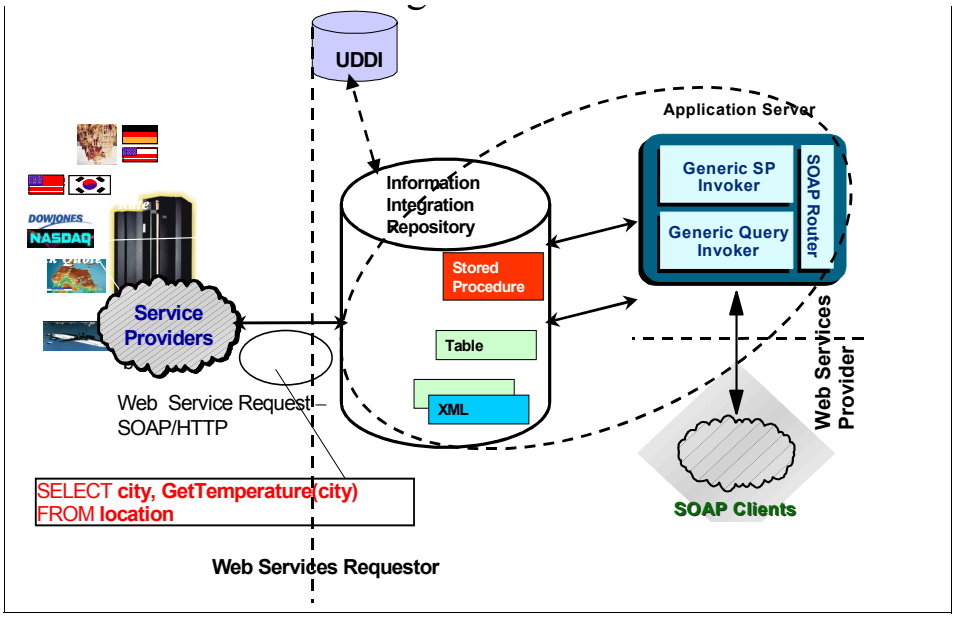


Figure 2-10 Information integration and Web Services

Structured relational and non-relational data, such as XML documents, may be exposed as a Web Service by the repository. Web Services can also be mapped to any SQL or XML operation by the repository, over heterogeneous, distributed data. The results may be provided as XML or SQL data.

As a requestor of Web Services, great flexibility is provided by the capability to integrate Web Services invocations with traditional requests for structured relational data (SQL statements). A single statement can access local and federated data as well as Web Services. Functional flexibility is also enhanced by providing support for generating SQL scalar and table User Defined Functions based on the description of the Web Service through its WSDL.

From an application development perspective, database application developers have traditionally built custom applications and “wrapped” the applications with a Web Services interface. Integration of information and data management functions into a Web application development environment such as the WSAD toolset greatly enhance productivity and ease the effort of creating Web Services using enterprise data.

Support for Web Services in the Information Integration component greatly enhances the ability of an enterprise to get a consistent logical view of enterprise information in a rapid manner through integration into the ESB discussed earlier.

This flexibility is critical for the rapid ‘time to value’ required by businesses in an on demand environment.

2.3 Methodology

It is evident that in any effort to enhance business flexibility there are likely to be a number of organizational functions and roles involved. In such a circumstance, even with a totally integrated technology environment, a number of organizational and governance challenges must be addressed in order to achieve rapid time to value. This is illustrated in Figure 2-11.

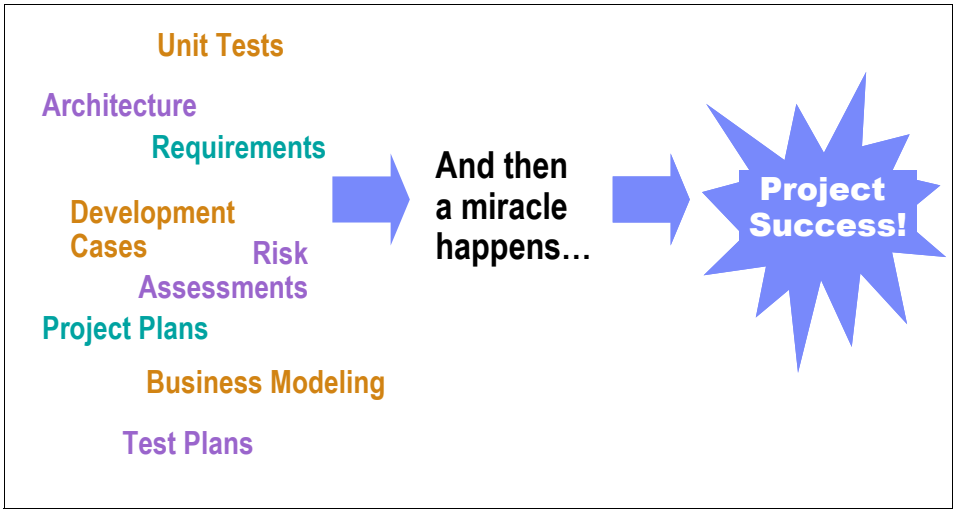


Figure 2-11 Need for a methodology

As the level of redesign becomes more sophisticated it is critical that an appropriate methodology be adopted in order to ensure that business results and time to value are achieved as expected. In fact, without a methodology, the likelihood of success simply with technology tools is significantly reduced, and in many cases the methodology is more critical to success than the technology toolset or run-time environment.

Given the objective of reacting rapidly to changing market conditions, an iterative approach to the project requirements with successive refinement is more likely to reduce time to value and risk than a traditional “waterfall” based approach. Figure 2-12 shows the approximate risk profiles of the two approaches:

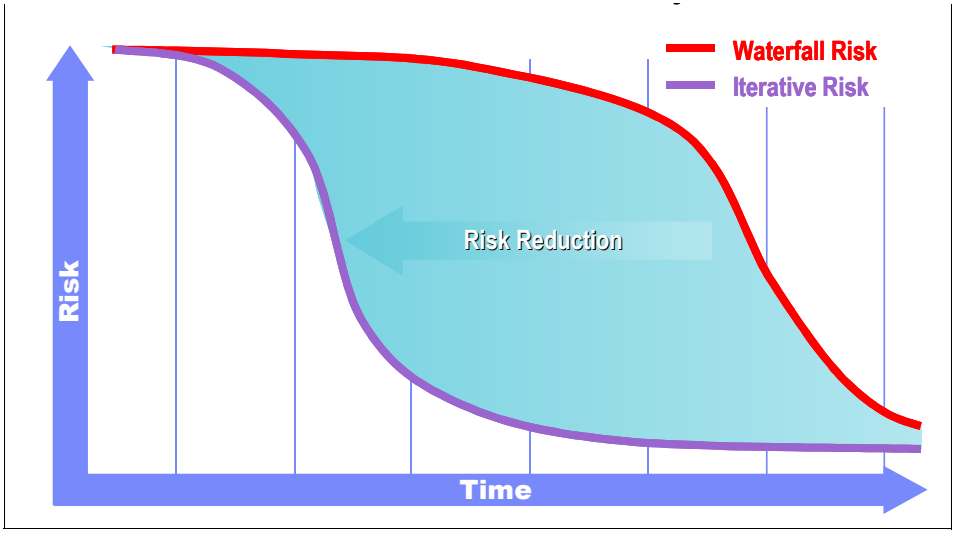


Figure 2-12 Risk profiles of two project approaches

Benefits of the iterative approach include:

- Risk mitigation in requirements analysis, architecture development, technology deployment, organization, and process.
- More effective change management, including functional changes, tactical changes, technology shifts.
- Learning as you go, which is critical to process improvement by providing a feedback loop. This approach also provides opportunities for enhancing individual learning.
- Identifies and promotes opportunities for reuse. In the traditional approach opportunities for reuse are often identified too late to be exploited and leveraged to reduce risk and time to value.
- Provides improved quality, since functions are tested earlier and more often.
- Provides greater predictability and minimizes losses, since any unexpected deviations from objectives are exposed early on, allowing decisions on cancellation to be made prior to significant investment.

Each iteration should be associated with a measurable business value. Typical iterative projects are likely to have the following phases:

- Inception
  - Define the vision and scope of the process implementation.
  - Make the business case.

  - Obtain executive support.
- Elaboration
  - Define/refine the detailed requirements for the implementation.
  - Define an initial process architecture with organization-specific content.
- Construction
  - Develop and plan pilot projects and refine as necessary.
- Transition
  - Roll out supporting tools to the organization.

As discussed earlier in this chapter, there are a number of roles involved in the redesign process. The level of involvement of each role changes with the phase of the project. Figure 2-13 illustrates qualitatively the involvement of various roles/disciplines in the organization in various phases of the project.

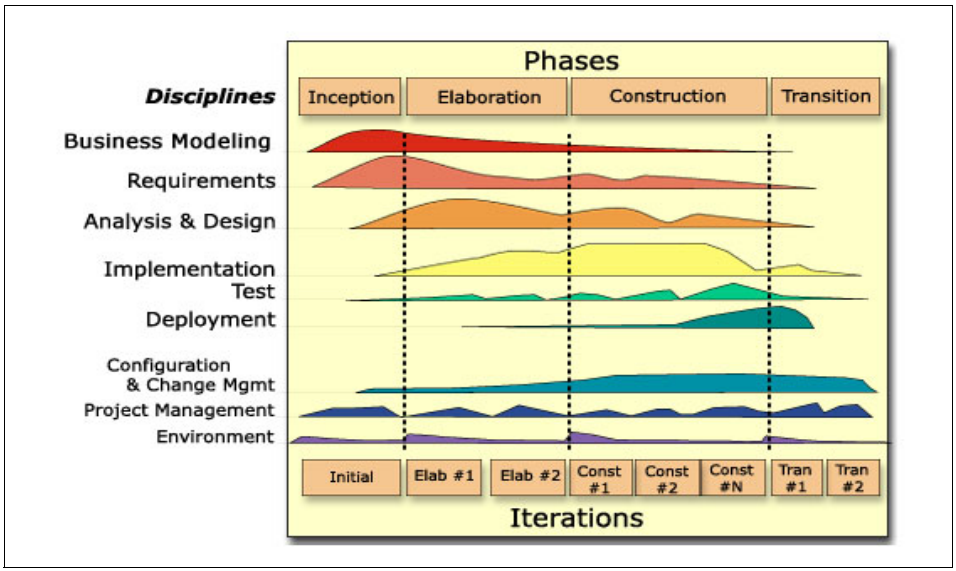


Figure 2-13 Iterative, incremental development

The use of tools that coordinate efforts of various roles through the project lifecycle and allow the cross-functional team to collaborate effectively provide significant productivity enhancements, while reducing risk and time to value. Some of the available tools are discussed in Chapter 4, “How to rapidly modify business processes” on page 63.

To effectively link the various functions together, an appropriate governance or organizational model is also necessary, as is strong executive sponsorship. In fact, the greater the number of functions involved in a project, the greater the need for strong executive sponsorship. One model of this includes a competency center approach. While a detailed discussion of such organizational models is outside the scope of this book, it may provide one reason why so far we have seen only a limited number of actual projects that utilize and leverage the end-to-end tool set described earlier.

2.4 Summary

As on demand technologies evolve, companies will increase their flexibility by taking advantage of open standards, Service Oriented Architectures and integration technologies. The ability to integrate people, processes, and information across an enterprise and its constituents is critical in today's competitive environment. Such integration will be driven by the requirements of the business and enabled through IT and business process solutions. Therefore, methodologies and governance policies must be in place to enable organizations to adopt new paradigms in how applications are built and supported by the IT environment. Though this vision of the future involves new ways of doing business, there are products and technologies available today to start building an on demand Operating Environment that can provide immediate benefits and position the enterprise to be more flexible and responsive in the future.

Part 2

How to’s for getting started

In this part of the redbook, we describe several approaches for getting started with deploying an on demand Operating Environment.

In Chapter 3, “How to achieve business flexibility through integration” on page 43, we provide several entry points based on common business requirements. These describe at a fairly high level how to address the business requirements within an on demand Operating Environment context.

In Chapter 4, “How to rapidly modify business processes” on page 63 and Chapter 5, “How to react in real time through seamless flow of information” on page 139, we describe two of these approaches in more detail and provide example scenarios of how an enterprise might plan for and deploy a solution today while keeping in mind and preparing for the longer term vision.

3

How to achieve business flexibility through integration

Most enterprises evaluating the on demand Operating Environment will consider a “start small” approach, based on an area or approach that addresses their current business imperatives or pain points. This is consistent with the vision of the on demand Operating Environment based on its modular technology framework and iterative methodology. It also allows enterprises to leverage their existing systems and skill base as they enhance their functions and capabilities.

This chapter describes starting points for adopting the appropriate portions of the on demand Operating Environment based on some potential customer-specific business imperatives. Each approach uses some, but not all, elements of the technology framework described in Chapter 2, “Integration overview” on page 11, in order to simplify getting started. Each of these approaches is described with how-to information. See the referenced sections for more details.

- 3.1, “How to simplify building, developing, and deploying on demand business applications” on page 44.
- 3.2, “How to improve communication and collaboration within and beyond the enterprise” on page 49.

- 3.3, “How to react quickly to changes in the marketplace by modifying business processes rapidly” on page 53.
- 3.4, “How to instrument applications and analyze events they generate to understand business process impacts” on page 55.
- 3.5, “How to create links between new and existing applications” on page 57.
- 3.6, “How to react in real-time by ensuring seamless flow of information” on page 60.

Each of these are discussed briefly in this chapter, providing a summary how to for getting started.

Two of them are described in greater detail in Chapter 4, “How to rapidly modify business processes” on page 63 and Chapter 5, “How to react in real time through seamless flow of information” on page 139.

Important: The choice of these two does not imply that they are any more viable or valuable than the others. We chose to provide examples with additional detail to demonstrate how any of these, or other, approaches could be expanded with more detail to plan and prepare for deployment.

3.1 How to simplify building, developing, and deploying on demand business applications

Businesses must be able to quickly define, develop, and deploy new business applications to react to changes in the marketplace. They must also reduce the cost and risks associated with application development and integration.

Whether used by a business analyst, IT solution architect, or application developer, this approach makes it easier to build, develop, and deploy on demand business applications. In fact, this approach is all about bringing these worlds together in the quest of making IT more of a “reconfigurable” reflection of business needs. This simplification covers more than just how business models can be accessed and executed, it is primarily about making it simpler to define and use those models.

3.1.1 Vision

Making this capability a reality requires integrated tools that support this effort from end-to-end while supporting best practices. In other words, it requires a united view of tooling for building, developing, and deploying on demand business applications. This includes high-level business process modeling tools,

Universal Modeling Language (UML) tools for modeling IT applications in a generic way, and IT-level tools used for specific environments. By adopting the appropriate tooling strategy, companies are able to integrate and extend their infrastructure to more quickly meet the requirements of the business.

Advanced business-level modeling tools will soon be extended to support the Business Process Execution Language (BPEL) OASIS standard, closing the gap between the line of business perspective and that of the IT infrastructure. This will expedite the extensions for business-level modeling beyond workflows by adding support for information entity modeling, organization models, policy modeling, and more.

This will allow for the adoption of Web-services-based process choreography and provide for interoperability between business process choreography, run times, and tools. These tools will deliver infrastructure workflow as well as complementary aspects of business process modeling. IT and business professionals will be able to seamlessly interact with business processes through “behind the scenes” use of UML-based modeling and run-time parameterization.

The evolution of these tools will include first class support for component-based application assembly and supporting patterns for business process application construction. This allows for the definition and re-use of business process best practices across an enterprise, and with its partners and suppliers, in an interoperable way.

From an application development perspective, data objects will provide disconnected and data-centric access capability to the application server.

Normalization of data access is key and will be accomplished through Data Objects and Web Services that extend federation capabilities. In particular, information integration capabilities will be made available as Web Services. These capabilities will facilitate the dynamic integration and delivery of information through enhanced discovery and federation of information sources. High-level tools for modeling of information integration will become more accessible, making the processes easier and faster.

User interaction will be focused around a single customizable paradigm. For instance, Lotus® Workplace will provide the framework that will deliver all aspects of collaboration built on a J2EE infrastructure. As a first proof point, the Lotus Workplace capabilities will allow for the connecting of people to business processes through a J2EE middleware tier. The Lotus portfolio of products will continue to be enhanced to provide an extended framework for supporting real-time collaboration such as Web conferences and instant messaging, as well as team-oriented spaces.

As business process best practices emerge across organizations and industries, businesses will be able to create solutions from canned process templates. With templates, instead of constructing applications from scratch, companies will only need to modify the templates to meet their unique business and infrastructure requirements. This will allow for a unified approach to process modeling and implementation of business process applications, providing tools focusing on business analysts, tools for IT architects and tools for developers. This unification will be accomplished via bi-directional bridges between these tools.

Access to necessary resources will continue to be simplified through an environment where users can interact dynamically with integrated business processes, peers, partners, suppliers, and customers, from their own role-based workspace. This will enable users to interact with business processes as part of a collaborative application and to generate templates based on commonly used applications and interactions for later reuse.

3.1.2 How to get started today

High-level modeling tools give business analysts the ability to design, test, and communicate complex business processes in their terms. With these tools, the business analyst can:

- Simulate the operational efficiency of processes and analyze potential business results.
- Simulate different business scenarios before they are implemented, allowing for the optimization of business processes.
- Include pre-built, deployable business processes and task definitions from a variety of run-time systems in process models.
- Capture important business processes and guidelines in one place for easy reference and company-wide consistency.
- Create model diagrams of current and new business processes, using a simplified graphical interface.
- Work in a collaborative development environment that enables on-line process design across functional teams in organizations.
- Define implementation templates in a variety of forms, including standard business process languages and modeling technologies.

Business process choreography starts with high-level business modeling, which is available today via the IBM WebSphere® Business Integration Modeler.

Universal Modeling Language (UML) is the industry standard for modeling IT applications. As an implementation-independent modeling language, it forms

conceptual models that describe the main elements of the solution, not the specific mapping to run times (as provided by the IT-level tools).

UML-based modeling tools help IT architects create business value by improving their software development processes. These tools refine the business-oriented model into something more IT-oriented, without regard to the specific IT environment or implementation requirements.

The IBM Rational® software development platform integrates software engineering best practices, tools, and services. This platform helps companies create and extend their infrastructure, visually understand and plan for enterprise architectures and core systems, manage risk and compliance, and develop better software solutions. This platform includes state-of-the-art tools that provide the capabilities to help build applications for an on demand Operating Environment. On demand attributes not only apply to a company's core business, but also to its supporting processes such as application development. For example, the software development tools should also support the integration of people, processes, and information by enabling better collaboration between developers and automating and tracking information about the development process and status.

The IBM Rational solution is based on best practices, iterative development, and tightly integrated tools. As a result, development teams—even distributed ones—can improve communication and leverage assets across the lifecycle. Rational tools mitigate “uncertainty” on software development projects. Line of business and project managers are empowered by a controlled, predictable, and reproducible work environment where assets are carefully managed, ensuring business continuity for development. Rational provides patterns for constructing UML-based models.

IT-level tools include process modeling, adapters and portlet toolkits, among others.

Process modeling tools

Process modeling tools coordinate long-lived activities that span multiple systems and work groups. They help enterprises avoid bottlenecks by automating and managing task list assignment. The result is an integrated solution that can replace ad hoc and error prone systems, which may have been developed in a piecemeal fashion over a period of time and are now expensive to maintain and enhance.

IT developers expect first class support for process choreography scripting. This allows for rapid process modifications and the simplified management of business logic. In addition, these modeling tools should allow for the separation of process logic that describes service choreography and the actual realization of

those activities. This is essential for supporting the dynamic modifications of process behavior and process “binding” to real service implementations. Process designers can choose to bind process activities at design time from a palette of available service implementations, or they can defer the selection of a service until it is needed at run time.

There is also a need for application development tools that provide a standards-based, service-oriented development, testing, and deployment environment. These would include tools and support for consuming services from and providing services to the Enterprise Service Bus. Also included are graphical business process composition tools that enable:

- Composition of a service out of one or more existing services
- Definition of the transformation of the flow of information between services
- Creation of a process that contains other nested processes

Adapters

Adapters for a number of popular packaged applications are available, including Customer Relationship Management, Supply Chain Management, Enterprise Resource Planning, and mainframe applications.

Application adapters extract data and transaction information from leading and industry-specific packaged applications.

e-business adapters provide proven solutions for securely connecting through firewalls to customers' desktops, to trading partners' internal applications, and to online marketplaces and exchanges.

Mainframe adapters leverage best-of-breed technology to access application data in OS/390® systems and provide connectivity approaches to AS/400® systems.

Technology adapters use a variety of protocols to access data.

Though there are many standard adapters available for the most common environments, integrated toolkits provide a framework for development of custom adapters.

These capabilities are made available through the WebSphere Business Integration portfolio of adapters.

Portals

Portals provide a mechanism for aggregating information and accessing enterprise services via a single consolidated view for Web usage. A portlet (similar to a servlet) provides access to a specific application or function being

made available to the user via the portal. Toolkits are available that enable you to customize and manage the enterprise portal and create, test, debug, and deploy individual portlets and Web content. Templates enable developers to quickly and easily create their own portlets. Debugging and deployment tools shorten the development cycle. Sample portlets that demonstrate best programming practices are also provided. The IBM Portal Toolkit Version 5.0 plugs into the IBM WebSphere Studio Workbench, which provides a comprehensive framework for the development of e-business applications.

In addition, there is tooling available for information integration that allows IT architects to define views on federated data. This is available as part of the IBM DB2 portfolio.

These IT-level tools are available today via the IBM WebSphere Studio Application Developer Integration Edition and the IBM WebSphere Process Modeler (shipped as part of IBM WebSphere Application Server Enterprise), IBM DB2 Development Center.

Access to applications, information, and collaboration capabilities is enabled through the IBM Lotus portfolio of products and portal technologies.

The following IBM products should be evaluated for their use in developing a solution using this approach:

- IBM WebSphere Business Integration Modeler
- IBM WebSphere Studio Application Developer Integration Edition
- IBM WebSphere Process Modeler
- IBM WebSphere Application Server Enterprise
- IBM WebSphere Business Integration Adapter Development Tools
- IBM WebSphere Business Integration Applications Adapters
- IBM WebSphere Portal
- IBM Lotus Workplace
- IBM Rational Rose®
- IBM Rational XDE™

3.2 How to improve communication and collaboration within and beyond the enterprise

On a daily basis, employees, customers, partners, and suppliers struggle to find and make use of the most accurate and relevant information available. Once that

information is found to have business value, users are looking for ways to communicate that information to others within and beyond their enterprise, to collaborate based on that information, and to make decisions based on it. In short, use it to impact the business.

This communication can be ad hoc or structured, and the participants can be internal or external to the business. Communication and collaboration is not only required and performed between people, but also with applications and data sources and repositories. Data sources encode business-relevant information and applications enable business logic that achieves the necessary communication and collaboration to achieve business goals.

The Service Oriented Architecture model normalizes the basic concepts of people or applications performing activities on information. Activities could include a person consuming data through a portal, or a CICS® application performing a transaction.

From a “people” viewpoint, services could be realized, as an example, by a portal-based workplace that enables interactions and by gateway components implementing B2B interactions with external business partners.

From a “process” point of view, the SOA provides a view where people, processes, and information are all represented as services. Existing applications and components can be exposed as services that other services can communicate and collaborate with.

Access to federated information provides the common view and data source transparency that enables people and processes to collaboratively act on business information.

3.2.1 Vision

The need for companies to respond faster continues to increase, as does the need to stay focused on the core business demands. Therefore, companies will remain dependent on the interaction between their employees, customers, partners, and suppliers. The need to develop and implement new ideas, share new information, and so on will be crucial to the success of departments and entire corporations.

For this kind of environment, collaboration capabilities will need to be available on demand. Business activity management workplaces, made up from portal elements, enable people to see and interact with business activities in which they are involved. For example, a worklist portlet lets people see what they need to do within a workflow or business process. Another example is a dashboard portlet that lets individuals monitor the performance of business processes or activities.

Process monitoring and analysis can enable employees (and to a certain extent partners) to participate in and control the execution of business process applications. This capability is enabled via “workplaces,” which create a way to provide and customize information for consumption by people, whether employees, customers, or partners. A unified view of the business activity will be enabled by a common event management infrastructure that unifies business- and system-events produced while executing business process applications.

Enhancements to information federation capabilities will allow for the inclusion of information from applications such as Siebel and Peoplesoft. This allows adapters to become information sources for federated views that can be created through automated discovery. This discovery could occur when new database tables are created, or when a change is made to the schema of an SAP business object.

Communication and collaboration will take on a new meaning when entire processes and new information is more readily available for users to access, analyze, and interpret. With enhanced template capabilities in Lotus Workplace, IBM partners have the ability to serve customized workplaces to their markets and also provide interaction with business processes to users.

Management of this activity will provide deep insight into business process execution from a high-level, business objectives perspective down to monitoring IT resource usage of specific process instances.

3.2.2 How to get started today

An approach to addressing this need for collaboration in the extended enterprise is to deploy solutions that allow employees, customers, partners, and suppliers to collaborate in ways that will lead to faster and more focused responses to changing dynamics in the marketplace.

This includes allowing multiple channels of communication such as:

- Instant messaging
- Sharing of moderately unstructured information through team rooms that can be set up rapidly to reflect the dynamic nature of teams
- Scripting of information and workflow within a workplace to reduce errors in the flow of information between people
- Pre-defined workplace components providing users with access to commonly used functions or to popular applications to allow rapid deployment of collaborative capabilities
- Role-based end user interfaces and simple scripting models for users to compose custom workplaces to enhance productivity

The IBM WebSphere Portal provides all of these collaboration and user interaction capabilities.

The human interaction aspect is only one component of improving communication and collaboration; the interactions of processes and information also play important roles.

Businesses can now integrate their business processes to extend the reach of workflow participants, using graphical Web-browser-based tools. In addition, businesses can connect business processes to applications via a myriad of outward facing interfaces through IBM's business connection and gateway products. These are available through the IBM WebSphere Process Modeler that is part of the WebSphere Studio Integration Edition.

These interactions generalize the inward facing notions of business process choreography to include business process connectivity with external partners and partner management. IBM provides a set of product offerings for connecting applications and systems across a company and to partners and customers. These connections allow businesses to:

- Reduce the cost of the value chain by utilizing a common, standards-based approach for accessing required services
- Maximize the investments in existing systems by revitalizing the use of the assets by exposing current applications and processes as services to enable B2B in a timely and cost-efficient manner
- Leverage industry interchange standards and protocols
- Streamline processes with partners and suppliers and better manage change due to shifting business conditions

Businesses also face the need to work with their data—stored in multiple locations and databases, used for multiple purposes, and accessed through existing applications—so that it can be used to improve communication and collaboration. Through the Common Information and Resource Model, companies have the ability to federate information from internal sources and remote partners or suppliers.

This information model enables people to define the information needed for their business process and link it with their existing data sources. Depending on the type of application, integration of this information is available through DB2 Information Integrator Global View or the WebSphere Business Integration Common Business Objects (CBOs) in combination with WebSphere Business Integration Application-specific Business Objects (AsBOs).

The Global View approach is more suitable for decision support and query-based applications, while CBOs and AsBOs are more suited to transactional

applications. AsBOs provide a common view of information hosted by existing applications such as ERP packages, while the CBOs provide an application-independent “domain model” for information relevant to a business in a specific domain. WebSphere Business Collaborations are business process templates that operate on CBOs and are “bound” to a real-world application environment by mapping CBOs to AsBOs.

The following IBM products should be evaluated for their use in developing a solution using this approach:

- IBM WebSphere Portal Server
- IBM WebSphere Business Integration Connect
- IBM WebSphere Business Integration Applications Adapters
- IBM WebSphere Studio Application Developer Integration Edition
- IBM WebSphere Web Services Gateway
- IBM DB2 Information Integrator
- IBM Lotus Workplace

3.3 How to react quickly to changes in the marketplace by modifying business processes rapidly

Continuous change is a given for most businesses competing to deliver customer value quickly and efficiently. While change is a constant, a company's need to remain focused and on track with its business is critical and requires the ability to modify business processes rapidly.

Through business process choreography, adding, deleting, and re-ordering these steps is made simple. In an open-standards framework based on a Service Oriented Architecture, it is also possible to dynamically replace various components of the process as business needs demand.

3.3.1 Vision

Technologies in the area of business process choreography will continue to evolve. Externalized business rules will be able to be modified without affecting other application components. Business rules that are externalized may be used in many processes; for instance, a business rule may include the criteria for what defines a preferred customer. These rules should be available to many business processes and applications, while remaining centrally controlled and maintained.

Standards, such as Business Process Execution Language (BPEL), will allow companies to model processes and communicate them through cross-enterprise processes that speak to other enterprises/partners. This will allow business processes to be linked across lines-of-business and even with other companies from a process choreography perspective.

Mediations, support for business process construction patterns, and more pre-defined solution elements (processes and process templates), will all be available and integrated into the Enterprise Service Bus. This allows more services and activities to plug into the overall business process more rapidly and extend the level of choice of implementation for those application components.

The development of process components will be enhanced by the way information is accessed and processed. More dynamic data replication, transformation and movement will allow more flexibility in the development and deployment of processes. This is enabled through information integration tools that understand and utilize business integration adapters.

Lotus Workplace provides the first step towards giving users the ability to interact with business processes in the space of collaborative applications. This is about complementing well-structured processes that involve people with the appropriate tools that those people need to perform the steps that are key to the overall process—for instance, ensuring that users have access to facilities such as Sametime® and team rooms so they can efficiently perform the collaboration required for the particular step of the business process.

Rapid modification of business processes will take on a new form as the model-driven approach to business process application construction evolves. Business process application development will become model driven, starting with the business-level process sketch, to process design, to process assembly. By the use of standards, development tools will be integrated and will support a set of patterns for constructing the processes. Over time, this becomes increasingly automated—from high-level modeling and design through implementation.

3.3.2 How to get started today

Refer to Chapter 4, “How to rapidly modify business processes” on page 63 for a more complete discussion of this approach.

3.4 How to instrument applications and analyze events they generate to understand business process impacts

The ability to monitor performance of a business process based on pre-defined business metrics is critical to accurate decision making and assessing the return on investment. This begins with an understanding and precise recognition of the business performance metrics that are used to evaluate the success of a process. In addition to business-level events, correlation of IT decisions to the business metrics provides critical information on future investments and IT strategy.

There are three important aspects to this approach. First, one must be able to instrument applications so that they can be measured accurately against business metrics. Second, there needs to be an infrastructure to handle the resulting events and direct them to the appropriate location or person for action. Third, appropriate actions must be taken based an analysis of these events.

3.4.1 Vision

A common event infrastructure (CEI) provides the umbrella for normalizing the format of business-level events produced by business process applications, and also IT-level events that might be related to business-level problems. Process integration provides the framework to instrument process implementations with consistent business- and IT-level monitoring instrumentation. This greatly simplifies the current environment where multiple events are triggered and formatted differently within the infrastructure.

Enhanced business process modeling and associated monitoring tools for end-to-end business process applications become available. Business Process Management (BPM) extends this ability to actually perform end-to-end metrics from modeling to application instrumentation and capturing resulting events as well as monitoring and analyzing events. These capabilities are expected to be provided in a standards-based J2EE infrastructure to leverage investment in skills.

All components of a business process application will produce common events using CEI. Business Activity Workplaces will be used for constructing, analyzing, and monitoring what goes on in the workflow. Toolkits can build dashboards and workplaces for more effective business management.

As business and system events become integrated, this will open the door for more autonomic management of business processes affecting the IT infrastructure that supports them.

3.4.2 How to get started today

As part of the business process modelling discussed earlier, the business analyst can set up business metrics for measuring effectiveness of the business process. Applications and the infrastructure are then “instrumented” to provide the appropriate information necessary for these measurements.

IBM WebSphere Business Integration defines the base metrics of the process. IBM WebSphere Process Modeler translates those into instrumentation of the actual application implementing the process.

Once this instrumentation is part of the business process, the infrastructure needs to be able to display real-time information from a variety of sources to allow decisive business performance management and optimization. Monitoring allows companies to:

- View work-in-process items and perform corrective actions by reassigning or suspending them
- Feed real-time data back into process modeling tools to deliver continuous business process improvement
- Base business decisions on actual performance data and resolve issues before they jeopardize business goals.

The results from this monitoring are most effectively displayed in the form of a dashboard that tracks processes, work items, and business performance metrics, and generates reports based on real-time and historical data that can be analyzed for trends and quartile analyses. This capability is provided by the IBM WebSphere Business Integration Monitor.

The following IBM products should be evaluated for their use in developing a solution based on this approach:

- IBM WebSphere Business Integration Monitor
- IBM WebSphere MQ Workflow
- IBM WebSphere Business Integration Message Broker
- IBM WebSphere Business Integration Collaborations
- IBM WebSphere InterChange Server
- IBM WebSphere Business Integration Server

3.5 How to create links between new and existing applications

All enterprises rely heavily on a variety of applications to manage different aspects of the business, both within their enterprise and connecting to external partners and suppliers. In many cases, an enterprise may have multiple CRM, ERP, or legacy applications across various geographies and business units. Businesses depend on these applications, regardless of whether they are existing or new, to make appropriate and timely business decisions on a daily basis. Businesses must leverage the investment and value in existing information and applications.

As new applications are added to an enterprise's infrastructure, maximum value can be obtained by ensuring that other facets of the business can take advantage of the related information to create new views of enterprise information, for instance, ensuring that a new accounts receivable application links to the existing CRM application or that the new purchase order application links to the SCM application. By failing to leverage existing applications and integrate them in with new ones, businesses increase their risk profile and time to value, and reduce their return on investment.

Enterprise business processes serve as a link between applications within the organization. These new and existing applications act as building blocks for the processes that they serve and the challenge is to make it possible to actually use existing applications in the context of new business processes.

As new applications are added to an enterprise's infrastructure, maximum value can be obtained by ensuring that other facets of the business can take advantage of the related information to create new views of enterprise information, for instance, ensuring that a new accounts receivable application links to the existing CRM application or that the new purchase order application links to the SCM application. By only deploying, but not linking new and existing applications, companies decrease their return on investment in that implementation by not taking full advantage of the efficiencies that result from an integration between applications.

3.5.1 Vision

The need for enterprises to link new and existing software applications will only become more important, especially as the amount of data and the need to track up-to-the-minute activities across multiple applications continues. Also, as businesses continue to focus on their core business and rely on an extended network of partners and suppliers, close integration with external applications and systems remains vital. The use of adapters will continue in the near-term,

but there will be extensions in the value of adapters to the overall integration. Adapters will become common and will be based on the Java Connector Architecture (JCA) standards, making it easier and faster to develop and implement new adapters for process integration and portal applications. Also, BPEL-based process choreography will serve as the pervasive model for scripting application integration. This provides a model for scripting business processes, information integration, and adapters in a consistent way.

Information federation will continue to expand beyond simple applications and portals to include additional support beyond information sources such as relational databases. For example, federation support for applications like SAP, Peoplesoft, Siebel and others using WebSphere Business Integration Adapters are enhanced to support federation of Web Services and WebSphere MQ.

As the amount of information and the number of applications grow, the need for enhanced collaboration with these applications and people increases. The Lotus Workplace framework leverages the adapters to provide collaborative Workplace applications that interact with third party software. This means that adapters to ERPs can be “plugged-in” to a Workplace application, allowing instant collaboration beyond enterprise walls.

Unified adapters make it easier than before to unify the different approaches for adapting to existing processes, enterprise information, and collaboration through portal technology. The end result is that people can use the same adapter to connect a process or activity to an application and to integrate the information hosted by the adapted application into federated views on enterprise information. They can also render information and tasks hosted by that application in a workplace. Common adapter toolkits make the construction and implementation of these adapters simpler, even as functionality is greatly enhanced.

In addition, not only will there be harmony between the adapter-based interaction with applications and information sources, but databases will also use the same “choreography” technology to perform data movement between information sources.

3.5.2 How to get started today

The ability to link applications within an enterprise and connect to external partners and suppliers is possible today through adapters and business connectors, connected through the Enterprise Service Bus described earlier. Application adapters extract data and transaction information from leading and industry-specific packaged applications. They enable access via application programming interfaces where possible. These adapters, such as mySAP.com, Siebel e-business Applications and Peoplesoft, provide bi-directional, real-time integration. e-business adapters provide proven solutions for securely

connecting over the firewall to customers' desktops, to trading partners' internal applications, and to on-line marketplaces and exchanges. Mainframe adapters leverage best-of-breed technology to access application data in OS/390 systems as well as provide connectivity approaches to AS/400. Technology adapters provide various ways to access data using protocols that enhance an integration infrastructure. These capabilities are made available through the WebSphere Business Integration portfolio of adapters.

IBM WebSphere Business Integration Connect V4.2 enables operational B2B based on communities of trading partners. It allows for the connection and integration with communities of trading partners, improved interactions with suppliers and customers, and defining trading partner interactions with an easy to use graphical interface, to rapidly enable and manage partners.

Information federation allows businesses to view, match, and compare data across multiple applications and databases. That single view greatly increases the ability to track activities related to a customer, an order, receivables, and more. This is possible using IBM DB2 Information Integrator.

Portals and their integration with other applications are a key component of an integration strategy that links people with applications. They allow for extended communication and collaboration between employees, customers, partners, and suppliers, to specified applications and information. Each of these parties, as appropriate, needs the ability to access certain applications and information rapidly and then collaborate based on that information with each of the interested parties. This is achieved through the use of adapters and federated views mentioned previously. Collaborative capabilities are provided through the IBM WebSphere Portal Server.

The following IBM products should be evaluated for their use in developing a solution based on this approach:

- WebSphere Portal Server
- WebSphere Process Modeler
- WebSphere Application Server Enterprise Edition
- WebSphere Business Integration Connect
- WebSphere Web Services Gateway
- WebSphere Business Integration Common Business Objects
- DB2 Information Integrator
- Lotus Workplace

3.6 How to react in real-time by ensuring seamless flow of information

There is a major difference between having relevant information and being able to access relevant information quickly in order to make business decisions in real time. On demand businesses must be able to find and realize the value of information, across business boundaries and independent of how it is stored.

Businesses must enable information to flow through the business processes within the enterprise. Information, in all forms, is critical to the continuation and completion of business processes to achieve objectives. Further, businesses must make this information accessible to employees, customers, partners and suppliers in a way that makes it simple to interpret and take action on in real time.

Enterprises have significant investment and value in the “legacy information” in an enterprise. Users require the flexibility to define what kind of information they are interested in, independent of where the “real” information is stored or what application generates and maintains it.

To ensure seamless flow of information, it should be easy to interpret. Fields such as date, time, name, and more can all be compared and viewed in the same way. With normalized data fields, analytics can be applied to that information to assist with more complex interpretation, like identifying patterns in the data.

3.6.1 Vision

Access to more sources, faster movement of information, and more caching are all part of what will allow more seamless flow of information. Several innovations will make this possible, including the use of adapters as information sources. This allows applications not only to connect to relational data sources, but also to ERP systems, legacy applications, and more.

Another aspect of seamless information flow is the ability to support the flow of unstructured information between people. Information sharing and support for collaboration greatly reduces the time it takes to complete tasks and steps in a process. This could be in the form of team rooms, instant messaging, or simple e-mail. More and more, e-workplaces will allow for greater collaboration due to increased access and ability to view enterprise information. With more information available to the right people at the right time, decision making is improved.

Developments in caching lead to more comprehensive and dynamic data placement management. Data provisioning, based on policies, is an important element of an on demand Operating Environment. Data placement management addresses how to optimize the initial placement of and access to all information

resources, as well as how to dynamically, and perhaps automatically, redeploy information resources to accommodate changing application and business requirements.

Information grids are built to support sharing of data and large-scale collaboration across heterogeneous platforms. Information grids provide information virtualization at the file, storage, and data levels.

There is a close connection between business processes and information integration. Business process integration needs access to federated data; which will be easier from an application modeling perspective since this is pushed into the information integration layer (and also more efficient from an execution perspective since it is compiled into the database).

Lotus Workplace ties all elements of collaboration directly to business processes. This allows the content from collaborations taking place throughout the process to be associated and transmitted with a business process, providing greater context. This greatly assists decision making by letting users later in the process know what was communicated prior to their interaction, cutting down on the amount of misinterpretation of information.

3.6.2 How to get started today

See Chapter 5, “How to react in real time through seamless flow of information” on page 139 for a detailed discussion of this approach.

3.7 Summary

This chapter has provided an overview of six different approaches or how to’s that an enterprise might use to start implementing an on demand Operating Environment today. These how to’s cover the various aspects of the integration component of the on demand Operating Environment. However, other approaches are also possible. We chose these as representative of the types of issues many enterprises may be facing today.

For each approach, we provided an overview of the business requirement, a vision of the type of functions that will be required and available in the future, and a short description of how one can get started today with products and capabilities that are already available.

Two of these how to’s are described in much more detail in the following chapters. There is no specific significance to the approaches we chose to detail in the following chapters. Rather, we just wanted to provide examples of how one might start the detailed planning and deployment of an on demand Operating Environment based on any of these approaches.

4

How to rapidly modify business processes

This chapter discusses an approach by which customers can leverage an on demand Operating Environment to react to changes in the marketplace and their customer requirements by modifying business processes in a rapid manner.

After introducing and positioning the technologies that provide this capability, this chapter describes two practical scenarios to highlight how these concepts can be applied to real world business requirements.

4.1 Introduction

Most businesses today are in an environment of continuous change driven by global pressures to provide customer value faster and more efficiently. In such an environment successful enterprises typically maintain focus on their core business competencies. This allows them to be flexible and adapt rapidly to changing business requirements. This ability to adapt includes the definition of new business processes and rapid modification of existing processes to allow higher levels of innovation and efficiency. Automation of manual steps, as well as monitoring and measurement of processes are key characteristics of such flexibility.

The capability to quickly modify business processes must be supported with toolsets that allow rapid formulation and evaluation of changes in business processes, coupled with the capability to reflect these process changes in the supporting information systems in a rapid manner.

The on demand Operating Environment framework is designed to allow the IT infrastructure to reflect these changes in business processes in a rapid manner.

Briefly, the components of the framework that apply in rapidly modifying business processes to react to market conditions are:

- **Business process execution:** This includes the tools and linkages to allow various users (business analysts, IT architects, and IT developers) to model business processes, and design, build, and deploy systems more effectively. It is a role-based approach with effective flow of information from one role to the other as appropriate for the specific business issue being addressed.
- **Enterprise Service Bus:** This provides a mechanism to connect the various applications and information sources in an enterprise in a manner that can be mapped to the business process being addressed. It is based on open standards and allows effective connectivity of applications both within an enterprise and outside it to partner systems.
- **Adapters:** The approach described in this section is an evolutionary one and facilitates leveraging of existing applications and assets in an enterprise. Adapters facilitate the exchange of information with legacy systems through the Enterprise Service Bus.
- **B2B connections:** Used to map any service to another service that exists outside the enterprise on any available transport channel. B2B connections also provide partner management functions to define the external services, service level agreements, and so on.
- **Common information and resource model:** Provides a consistent way of accessing information across the enterprise.

4.2 General strategy

The strategy to allow an enterprise to rapidly modify business processes is based on the following elements:

- It represents an evolutionary approach that allows an enterprise to get started today and build on their experience and expertise.
- It leverages investment in existing systems and applications.
- It leverages the skills and roles in an organization required to effectively transform business processes through information technology. It provides the tools necessary for each role, from business analyst and IT architect to developer, with effective linkages between them.
- It provides for performance measurements based on business metrics.
- It is based on open standards to allow greater flexibility and connectivity to systems within the enterprise and outside the enterprise, such as with business partners.
- It decouples the business process from the IT infrastructure to minimize the impact of change in one on the other.
- It provides a business-service-oriented view of the IT resources in an organization.
- It provides loose coupling between various systems, to allow for scalability.

In addition to the technology itself, several other factors are key to the success an enterprise could potentially achieve through this approach. These include:

- An effective governance model that brings together the line of business and IT functions in an enterprise to ensure rapid response in this environment.
- Adoption of an appropriate methodology that allows effective use of the tools in the organization.
- Development of the appropriate skills, which can often involve a combination of business process and technology skills.

4.3 Solution components

This section provides more detail information on solution components for implementing this approach.

4.3.1 Business process execution

Business process execution defines how processes involving people and applications interact together and with various other resources to effectively and efficiently complete a business process.

Simple and rapid modeling of business processes not only saves time for business architects and technology specialists, but also allows a level of control for line of business managers based on their insights into the market itself. This simplification stems from the notion that business process execution separates process modeling from application components. This in turn enables process modification without affecting other components.

These capabilities are based on the availability of an integrated set of tools that can be used by various functions and roles in an organization and yet provide a seamless flow of information between them.

Broad categories of current customer adoption of redesign are shown in Figure 4-1.

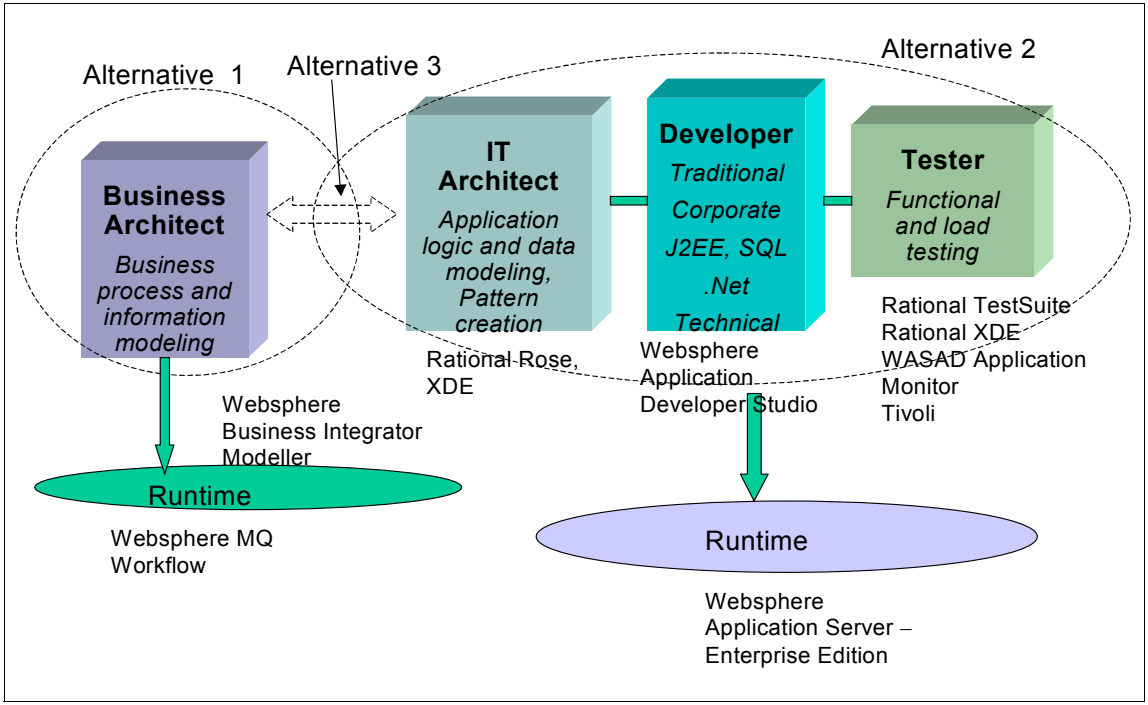


Figure 4-1 Current approaches to redesign: Role-based

As illustrated in Figure 4-1, customers tend to use three broad but distinct approaches to redesign. These are:

- **Alternative 1:** Typically led by the line of business and business analysts.
- **Alternative 2:** Typically led by the IT function in an organization.
- **Alternative 3:** This represents a more comprehensive approach that encompasses the entire tool suite and functional roles. Currently it appears to be used only in the most complex of redesign projects. Some customers have indicated that they are evaluating this alternative. In addition to the value, they are also considering the methodology and governance models with this approach, since they would be critical for its success.

Alternative 1: Led by business analysts

In many cases, a line of business function may choose to initiate a re-evaluation and re-engineering of their processes based on business objectives, such as the launch of a new product or service, or providing more efficient service.

The benefits of a tool to automate and speed up the modelling and deployment in such an environment are clear. In addition, it also could provide a valuable link for communications between the line of business and the IT departments.

Some characteristics and benefits of the modelling process are illustrated in Figure 4-2.

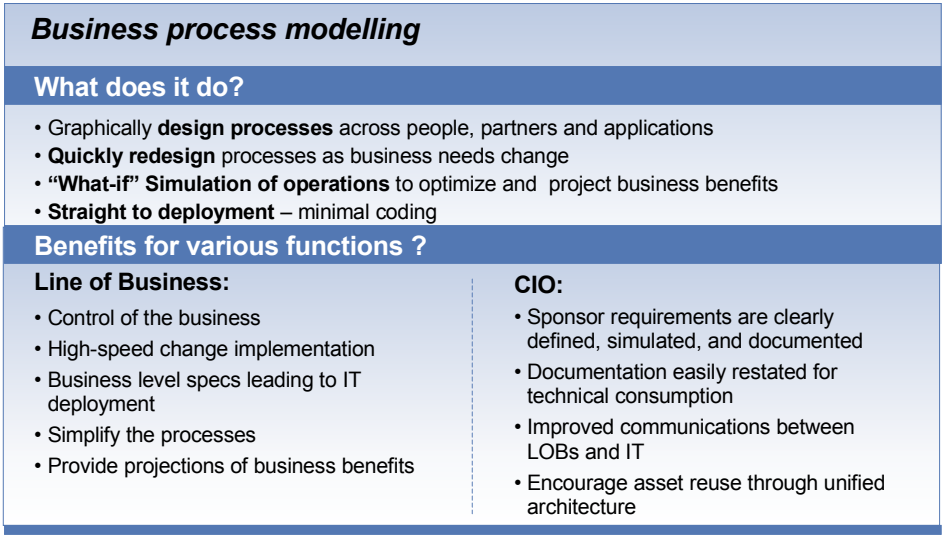


Figure 4-2 Business process modelling

Business process execution allows the analyst to achieve several objectives in such a scenario. The analyst can:

- Simulate the operational efficiency of processes and analyze potential benefits
- Include pre-built, deployable business process/task definitions from a variety of run-time systems in a business process model
- Capture important business processes and guidelines in one place for easy reference and enterprise-wide consistency
- Model diagrams of current and new business processes, using a simplified graphical interface
- Provide a collaborative development environment that enables on-line process design across functional teams
- Provide the means for the definition of implementation templates in a variety of forms, including standard business processing languages and modelling technologies

The business analyst does this by:

- Establishing a process modelling methodology
- Modelling the business process as it currently exists, including people and applications
- Creating the new business process; modelling and projecting various alternatives around it
- Defining business metrics to be measured
- Deploying the model and monitoring the business performance

In many cases, the line of business may choose to directly deploy the re-engineered business process to an IT technology infrastructure, minimizing the need for additional application development resources. Current users of this approach tend to be agnostic to the underlying technology infrastructure.

In situations where there is a need to integrate additional applications or develop new applications, this approach by itself may not be sufficient and integration with other development tools might be necessary. Additionally, if deployment to an infrastructure other than MQ Workflow is required, this approach would need to be integrated with IT tools and functions. While such integration appeared to be of interest to the customers who were using this approach, it did not appear to be a high priority and appeared to be in the research and evaluation phase. An overall methodology and governance model for use of an integrated toolset was also a key issue expressed by this customer group.

Currently, the capabilities described in this approach are provided by the IBM WebSphere Business Integration Modeler, as illustrated in Figure 4-3.

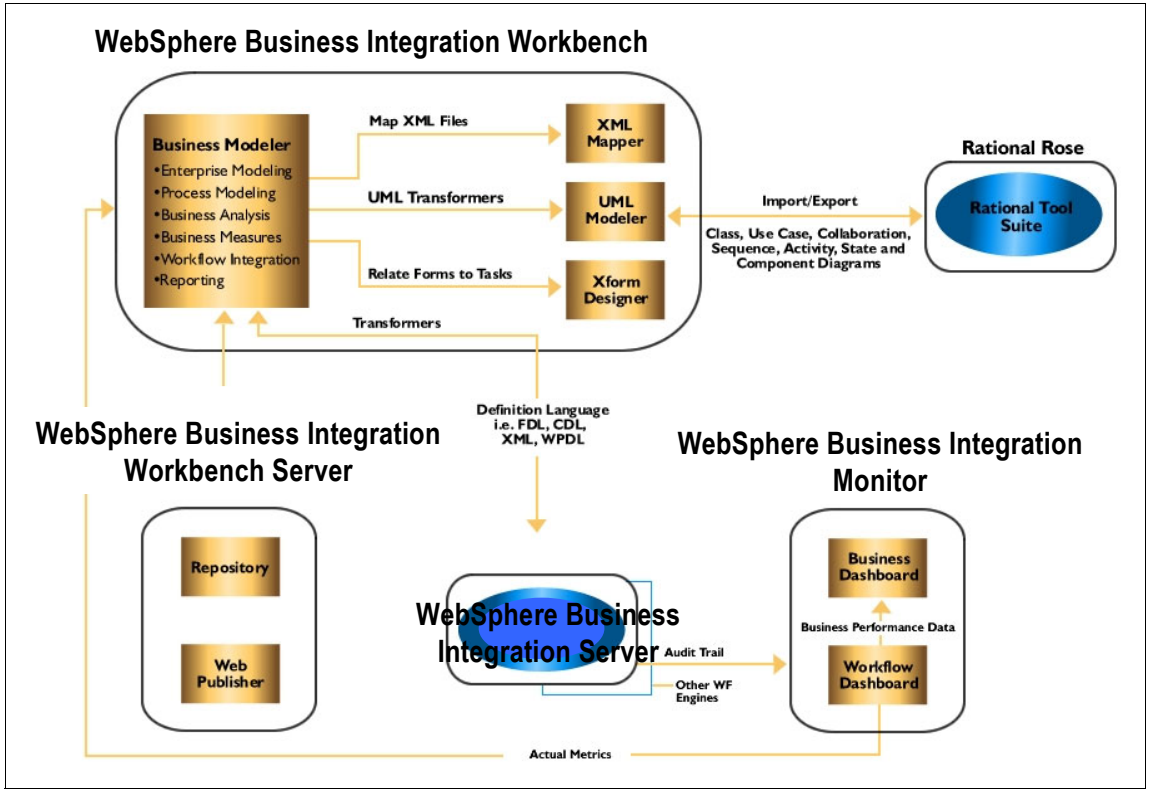


Figure 4-3 WebSphere Business Integration Modeler

As seen in Figure 4-3, IBM WebSphere Business Integration Modeler provides choreography for all kinds of business processes or flows. These business processes can involve a variety of services and resources.

WebSphere Business Integration Modeler provides a business-analyst-oriented approach. It includes management tools and is designed to help business analysts model and manage enterprise-level business processes using their business knowledge. They can monitor the business processes in real time. The modeler provides business process load simulation and costing facilities to refine models prior to implementation, and allows geographically dispersed sites to update and manage a single business process repository as well.

WebSphere MQ Workflow combines support for people-oriented workflow and fully automated application-to-application workflow in a single product with common interfaces.

WBI Modeler has the capability to generate Flow Definition Language (FDL) constructs, which can be deployed directly on the IBM WebSphere MQ Workflow server run-time environment. This allows an enterprise to rapidly integrate resources—people and applications—to support re-engineered business processes, with minimal IT development involvement.

Refer to later sections in this chapter for a discussion of how the WBI Modeler tool is expected to evolve to provide better integration with other standards-based modelling and run-time environments, as well as IBM toolsets focused on other IT roles in the enterprise.

This approach is also discussed in the context of a customer scenario later in this chapter.

There are several considerations to keep in mind with this option:

Advantages:

- This option focuses on a business process model rather than the IT model or development. It represents a more direct correlation to the business process.
- It includes both people and applications in the workflow context and business processes.
- It allows measurement of effectiveness and efficiency based on business metrics.
- It can be directly deployed to a run-time environment with minimal need for application development.

Disadvantages:

- Currently this option supports only the MQ workflow environment. As discussed later in this section, it is expected to be more comprehensive and open standards-based in the future.
- For customers without an MQ workflow run-time environment, additional investment in systems and skills is required.
- This option provides less IT flexibility, and does not address key issues such as how applications are integrated and new application development.

Based on these trade-offs, this alternative is typically used in the following customer environments:

- Line-of-business-driven modeling with minimum IT resources.
- Modeling complex workflow with human interaction.
- Organizations not focused on application development, and agnostic to the IT infrastructure requirements.

IBM WebSphere Business Integration Modeler does the following:

- Provides tools to design, test, and communicate complex business processes

- Simulates the operational efficiency of processes and analyzes potential business results
- Allows for the inclusion of collaborations from the WebSphere Business Integration Server into process models and to edit objects from within the WebSphere Business Integration Toolset - System Manager
- Captures important business processes and guidelines in one place for easy reference and company-wide consistency
- Helps to model diagrams of current and new business processes, using a simplified graphical interface
- Simulates different business scenarios before they are implemented
- Bundles the WebSphere Business Integration Workbench and the WebSphere Business Integration Workbench Server to deliver a comprehensive development and collaboration environment

Alternative 2: Led by IT

In a number of cases the systems redesign is led by the IT function once business requirements have been defined. Business objectives may not require complex process re-engineering or modelling, or the business analysts may feel that they have a sufficiently good insight into the business processes without the use of a tool. New applications and application integration techniques may need to be developed and projects are funded as such. Many customers that use this approach also have an open standards based run-time infrastructure on which they choose to deploy the applications and support business processes.

Another challenge currently faced by customers is the growing and increasingly scattered business logic and application data throughout the organization and across multiple software assets. Much of it resides in databases, packaged applications (such as ERP systems), or back-end systems (such as IBM CICS). Other business logic can be found in existing Java and J2EE applications. With the rise of open standards, business logic and application data will soon be available through internally and externally available Web Services. Instead of reinventing the wheel with every new application that is built, companies need a way to reuse their existing software assets and leverage the power of Web Services in the development of new J2EE-based applications.

With this approach, IT developers typically use tools to translate high-level process models into reality and also to script more fine-grained processes that realize the business logic one level below where business-level managers operate. By utilizing this capability to visually choreograph the interactions between various software assets, developer productivity is significantly enhanced.

This spans modeling of activities across multiple systems and workforce groups. In addition, enterprises can avoid bottlenecks by automating and managing task list assignment. This is a marked improvement over conventional ad hoc alert e-mails sent to system management professionals at all hours of the day and night!

IT developers define what kind of service they need to perform specific functions or activities in the process. In many cases, the developers will pick a particular service to accomplish the required task. Alternatively, the decision of which service to utilize could be made during the deployment of the application. This allows for an appropriate choice to be made based on the availability of services at the time of deployment.

Furthermore, companies are able to modify business processes more rapidly since underlying components (applications, databases, portals, and other resources) can all be modified or exchanged with minimum impact. Service requesters are decoupled from providers and flexible “re-wiring” of component interactions as needed is enabled with minimum impact on the rest of the process implementation.

Developers gain a standard way of representing and interacting with virtually any software asset and therefore don't have to spend time working with unique interfaces and low-level APIs. Drag and drop tools allow developers to define the sequence and flow of information between software assets. Individual software assets and even larger application workflows become building blocks that can be reused in developing other applications. Productivity gains continue as the run-time support for these new J2EE workflow capabilities is fully integrated with the application server, delivering a single administration and deployment environment.

In this alternative, more than one tool-set may be used. However, all the roles are generally within the IT function of an organization. Figure 4-4 on page 73 shows the requirements of the IT organization in more detail for such a redesign effort.

Role	Function	Skills and usage
Business Analyst		Non technical focus, Comfortable with business architecture, business modeling, and business processes
IT Architect		Highly skilled, modeling applications and information architectures, creates & manages reusable patterns & models
Developer	Traditional	Experienced in Cobol, RPG, IMS, CICS, little J2EE skill
	Corporate	Comfortable with VB / PowerBuilder, little J2EE skill
	J2EE	Experienced with J2EE, Web Services and XML technologies
	.Net	Experienced with Microsoft, Web Services and XML technologies
	Technical	Experienced building technical or embedded systems, C/C++ and/or Java
	Database / Info. Integration	Experienced with database and XML schema design, stored procedures, O/R mapping, conceptual to physical DB mapping
Test / Perf.		QE responsible for functional and load testing

Figure 4-4 Roles and usage

Several toolsets based on roles are provided by IBM for system redesign. Specific toolsets for the various IT roles in an organization allow the users to accomplish their objectives more efficiently. Some of these are shown in Figure 4-5 on page 74.

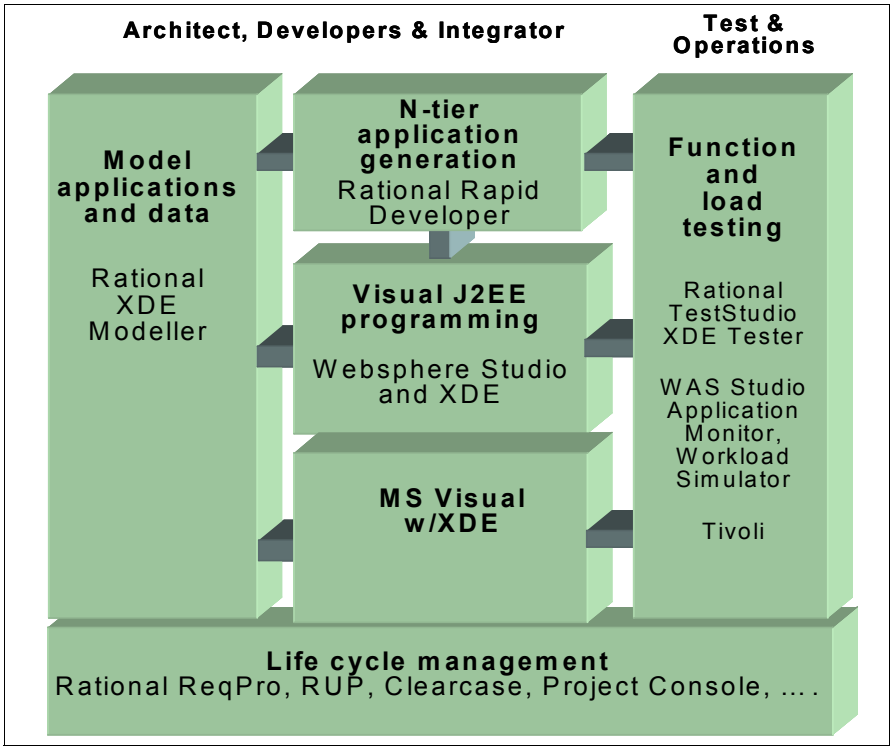


Figure 4-5 Some IBM toolsets for various IT roles

The current development toolsets in such an environment include:

- Rational XDE
- WebSphere Application Developer Integration Edition (WSAD-IE)

The WSAD-IE toolset is currently fully integrated into the open Eclipse environment. There is plug-in level integration for the Rational XDE environment into the Eclipse environment. This provides a relatively seamless user experience, as shown in Figure 4-6 on page 75.

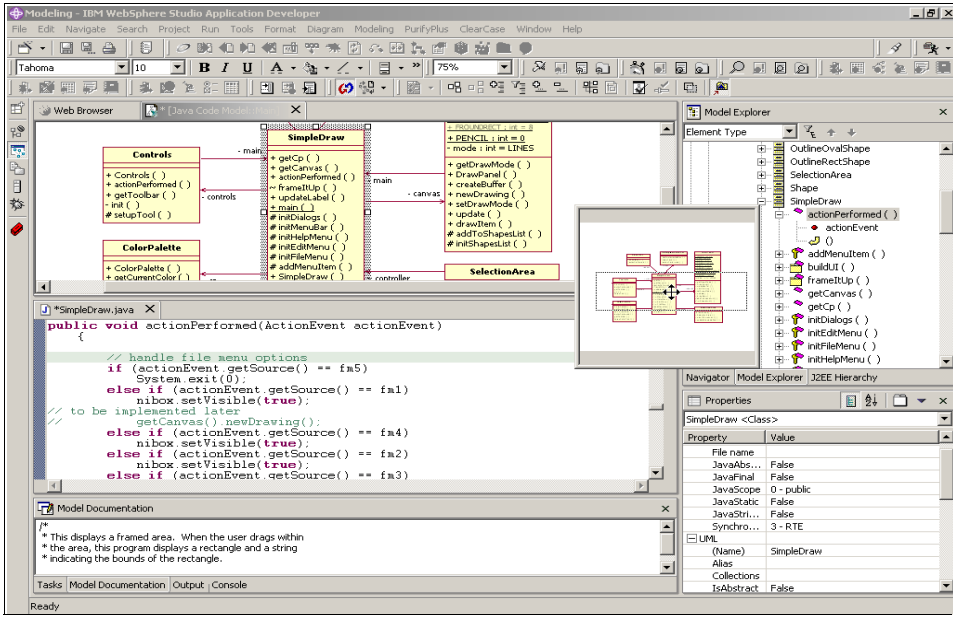


Figure 4-6 XDE and WebSphere Studio integration

An integrated development environment with Rational XDE and WSAD-IE provides IT architects and developers with the following features and capabilities:

- Relatively seamless user experience
- Full life-cycle modeling from an IT perspective
- Business and system use cases, realizations (use case, sequence, activity diagrams)
- Domain analysis
- Architecture, detailed implementation design
- Data modeling (E-R modeling using UML notation)
- Code modeling (Round Trip Engineering, automatic or on-demand code-model synchronization)
- Deployment modeling: model complex (multi-platform) deployment scenarios, synchronize with WSAD deployment descriptors, deploy directly from diagrams
- Conceptual IT models, not dependent on code
- Facilitate reuse of assets through:
  - Patterns engine
  - Use of asset repositories for browsing, importing, and exporting of assets

- Model-to-model transforms
- Code generation with Code Templates
- A foundation for model-driven development

While there is currently a significant level of integration between the XDE and WSAD environments, there are several areas in which the toolsets are distinct. These include:

- XDE is suited for model-centric development while WSAD is more code-centric.
- There is not a smooth unified edit-compile-debug process between the tools.
- There is limited sharing of artifacts between XDE and WSAD.
- There is not a single cohesive deployment mechanism that is used by both tools.
- Application level trace capabilities are not the same.

These integration issues are expected to be addressed in future releases of the toolset.

The run-time environment for this alternative includes:

- WebSphere Application Server Enterprise Edition. This provides the environment for a services-oriented architecture and business process execution based on open standards.

Figure 4-7 illustrates the sequence of activity from development on WSAD to deployment on a WebSphere Application Server.

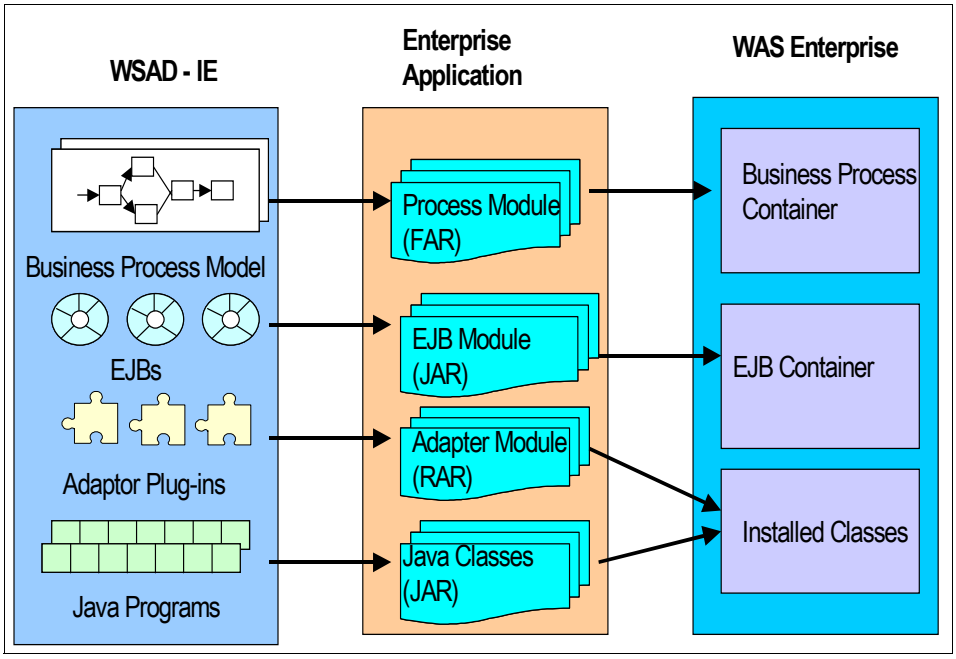


Figure 4-7 Business process execution using WSAD IE and WAS-E

WSAD-IE provides development, testing, and debugging tools for the business process flows. The development time component for developing flows is known as the Process Choreographer.

WAS-E provides the J2EE run-time engine of the workflow component. This run-time workflow component is known as the Business Process Engine.

Once a developer has created services out of an organization's software assets, the next logical step is to use those assets as part of a business process. Integrated J2EE workflow capabilities offer developers intuitive, flow-based development tools, to take existing software assets and quickly define how those assets are used within a J2EE-based application.

WebSphere Application Server Enterprise Process Choreographer is a component of IBM WebSphere Application Server Enterprise that provides run-time workflow functionality as an extension to the WebSphere Application Server. It delivers integrated J2EE workflow.

Refer to the later sections in this chapter for a discussion on how these IT-role-oriented tools are expected to evolve to provide better integration with business-process-oriented tools.

This approach is also discussed in the context of a customer scenario later in this chapter.

There are several considerations to keep in mind with this alternative:

Advantages:

- This alternative provides an IT-driven modeling approach, and provides productivity enhancements when new applications are to be developed or integrated, as well as throughout the IT lifecycle.
- It represents a transaction-oriented approach and is effective for systems that do not involve complex business processes and workflow.
- The deployment is on an open-standards-based run-time environment, which allows more flexibility for integration with other systems both within and outside the enterprise.
- The skills and roles required are generally contained in one function (IT) within an organization, simplifying the governance issues.

Disadvantages:

- This approach is IT-intensive and requires appropriate skills for implementation.
- It is not oriented towards business metrics and the correlation to business objectives is complex at best.
- Deployment with this approach is more complex and involves more steps than discussed earlier.
- Linkages and communications with line of business units may not be effective and may not be enhanced as a result of using this approach.

Based on these trade-offs, this alternative is typically used in customer environments with the following characteristics:

- Very limited modeling of business processes is required. The business analysts may already understand the business process and alternatives well enough that a process analysis tool is not required.
- New application development, integration, or application modernization are required to implement business processes.
- An open-standards-based run-time infrastructure is employed.
- Micro-flows oriented: application and application communication.
- A programming and IT architecture skill base exists.

Alternative 3: End-to-end modelling and redesign

This approach is currently under consideration by several customers and users that are evaluating significant business process and IT redesign efforts.

Based on the current customer environments, governance structures that include line of business and IT functions tend to be formed only when there is a significant enterprise-level project involved, such as the implementation of an ERP system. These clearly are infrequent events, requiring significant resources, and are not suitable models or patterns for a rapid and flexible organizational structure based on current practice. In the absence of such significant enterprise-wide initiatives with a strong sponsoring executive, the objectives of the business analyst community and IT community have been relatively well segmented.

The business analyst community has typically been more technology infrastructure agnostic than their IT organization counterparts. Similarly, the IT organization has typically focused on micro-level flows and such that are well beneath the focus level of most business analysts.

The toolsets used by business analysts and IT practitioners in an organization currently have different interfaces and only a rudimentary level of integration. Figure 4-8 illustrates this.

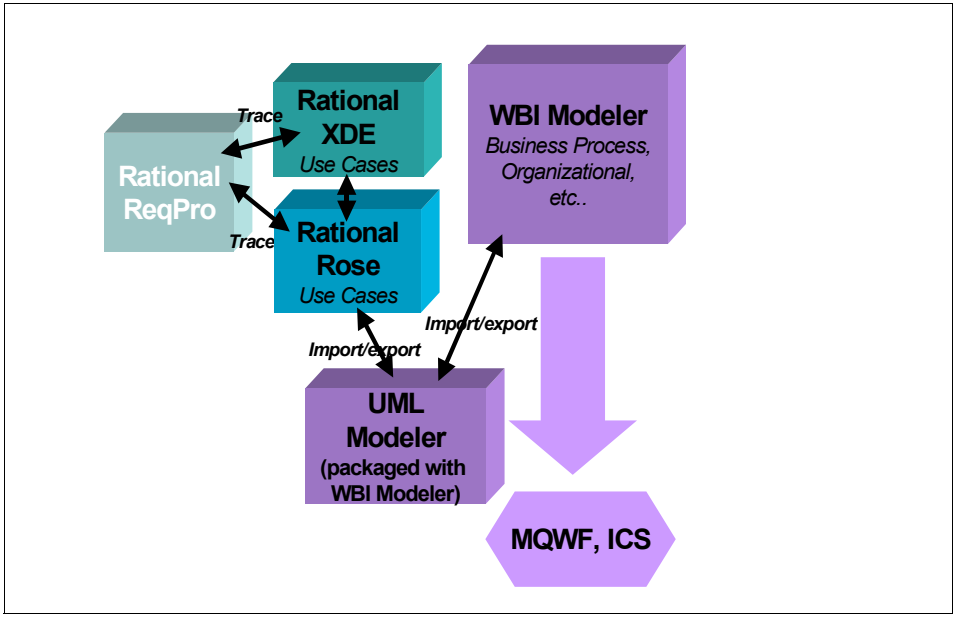


Figure 4-8 Current level of integration between business analysis and IT tools

Several areas of integration are planned for future releases of these tools. The following features and capabilities are in development:

- Easy to use flow and consistent user experience
- The ability to reconcile business model with implementation

- Consistent semantic relationships between business model and use cases
- Association of business model with an IT requirement
- Traceability of functions
- Version control of business models

Standards and toolsets are evolving to make the integration of the business process and IT role oriented toolsets more seamless, flexible, and open standards based.

In addition to the tool technology, customers and users are also evaluating governance models that will bring IT and line of business functions closer, as well as methodologies to implement such projects in a rapid manner. These aspects are considered critical for the success of integrated toolsets in an enterprise environment.

Further details on future developments of this integration and potential methodologies are included later in this chapter. The capabilities will be provided in an evolutionary manner so that the tools used by various roles in a business organization and the investment in skills is preserved and leveraged.

There are several considerations to keep in mind with this alternative:

Advantages:

- This is a comprehensive approach that supports business model redesign to programmatic construction.
- This approach allows open standards based infrastructures and Services Oriented Architecture (SOA).
- Facilitates knowledge transfer from the business analysis team to the IT architect and development team. Manages business process and IT knowledge at an enterprise level.
- Supports complex technology infrastructures.
- Helps establish and facilitate linkages between the business units and IT.

Disadvantages:

- Requires significant investment in skills.
- This approach needs a more complex organizational support structure.
- Lack of end-to-end methodology, encompassing current toolsets.
- Integration among toolsets still under development.

Based on these trade-offs, this alternative could typically be used in the following customer environments:

- Large organizations with strong executive sponsorship.
- Organizations with significant complex, re-engineering efforts underway that require new applications/integration to support the re-engineered business processes.
- Organizations with a significant skill base and resources.
- Organizations that are building a foundation for future systems development.

For detailed information on applying Rational tools to a simple J2EE-based project, visit the following Web site:

http://www.ibm.com/developerworks/rational/library/184.html

4.3.2 Enterprise Service Bus

Enterprises IT departments are constantly under pressure to deliver integration projects more quickly and on tighter budgets. This has never been more true than in the current business climate. Diverse platforms, execution environments, and programming models have all contributed to the complexity that often defeats the best integration teams.

But help may be on its way as middleware vendors focus on developing and implementing a new generation of infrastructure which is standards-based and focused on the service-oriented approach. By eliminating dependencies between clients and service providers, the Enterprise Service Bus promises to reduce complexity and pave the way for less expensive integration of application systems.

Service-oriented integration

The service-oriented approach (see Figure 4-9 on page 82) encourages systems architects and implementers to think about composing strategic new and legacy business functions as services. New functions can be built economically by reusing certain services and developing those that do not already exist. When successful, this achieves both improved development productivity—through reuse—and improved time-to-market.

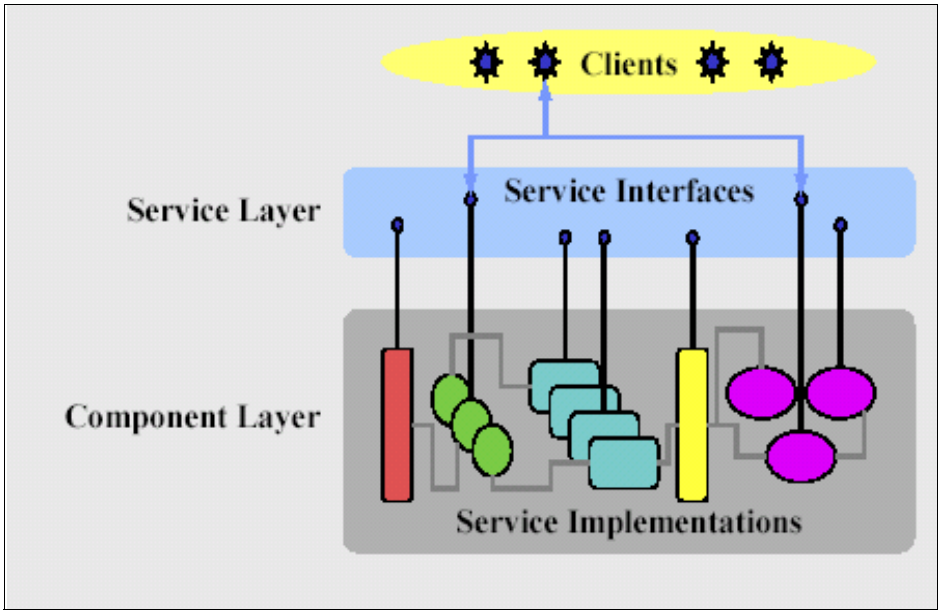


Figure 4-9 Service layer

Many enterprises claim to have implemented service-oriented systems in the past; some with outstanding success and some with mixed success. The difference between those projects and the current initiative is that:

- The new approach is based on open industry standards.
- It applies to both intra- and inter-enterprise domains.
- Software vendors are in full support, making available a variety of service-oriented tools and emerging middleware.

If this direction is maintained, with evolving standards and corresponding inter-operating implementations for diverse systems platforms, enterprises will benefit from:

- A growing pool of employees with the appropriate skills, and familiar with the relevant standards and technologies.
- Facilities for exposing standard service interfaces for, and reuse of, business functions independent of the platform they run on.
- Flexibility to offer services both to internal users and external users (suppliers, business partners, and customers) in like manner.

However, before leaping to the conclusion that the service-oriented approach has completely satisfied the seemingly impossible requirement currently placed

on IT departments, we must look more closely at the proposed approach and how the middleware infrastructure is evolving.

Decoupling to reduce complexity

The essential separation between the service layer containing service interfaces and the component layer containing implementations is one degree of freedom required to deliver IT flexibility and responsiveness.

By developing standards-based technology and implementing it on a wide range of system platforms, middleware vendors are removing the dependency between clients and services running in a network of heterogeneous nodes. The standards-based approach should remove dependencies on operating system, execution environment, and implementation language technologies.

By publishing standards-based service descriptions, the contract between client and services should also be free of any dependency upon implementation technology, as noted previously. The chosen standard, WSDL facilitates the definition of the information model, messages, and operations that are the basis of abstract interface contracts between clients and service providers. Other standards will describe the qualities of service provided with usage. For more details on these standards see the following Web site:

http://www.w3c.org

The format of messages exchanged in a service network is defined using the XML standards and the packaging of service request and response information defined by SOAP and other related standards. For example, SOAP defines a header for contextual information such as security signatures, and a body for business-specific information with attachments as required.

The protocol used for transporting XML standard service messages between clients and service providers is also removed as a dependency by the use of HTTP between Internet nodes. However, this protocol does not satisfy all requirements and so the standards approach now adopted is to eliminate this dependency by defining the choice of protocol—WSDL binding information—to be a dynamic or deployment-time issue.

The location of end-points in a service network is addressed by WSDL port information and so, like choice of protocol, this is eliminated as a dependency between clients and service providers since it relies on dynamic or deployment-time selection. The mechanism for discovery of port and binding choices available in WSDL documents is also defined by standards—UDDI and WSIL—to support deployment and run-time use of internal and external service registries by middleware. For more information, see the following Web site:

http://www.oasis-open.org

How decoupling will improve the business

If the principles summarized in the previous paragraphs are complied with, we get some interesting benefits:

1. There is real synchronization between the business and IT implementation perspectives. This has been a source of concern for many years: that the business people really don't understand the IT architecture and applications (not just in terms of functionality, but in terms of management issues, cost implications, upgrade policies, and so on). With well designed services we can radically improve communications with the business, and indeed move beyond alignment and seriously consider convergence of business and IT processes.
2. A well formed service provides us with a unit of management that relates to business usage.
3. Because the service is abstracted away from the implementation, it is possible to consider various alternative options for delivery and collaboration models. No one expects that core enterprise applications will be acquired as services from multiple sources, at any stage in the future. However it is entirely realistic to assume that certain services will be acquired from external sources because it is more appropriate to acquire them.

Service container middleware

When application server products, such as IBM WebSphere Application Server, fully implement the latest J2EE 1.4 specifications, service implementations will be deployed in containers along with other application components, with all the advantages of shared resource and management facilities. Multiple services with their interface definitions, ports, and bindings may be deployed in each container.

In fact, many of those existing application components will be given service WSDL descriptions for access as services but run otherwise unchanged.

In J2EE systems, services may be implemented as JavaBeans, Servlets, Enterprise JavaBeans, process flows and Java Resource Adapters – the latter providing access to legacy information systems, messaging backbones, and off-the-shelf software packages.

Service requests will enter containers from a variety of ports using a variety of different bindings—as described by the deployed WSDL documents for services provided—after passing through a "pipeline" of handlers (see Figure 4-10 on page 85). These handlers will provide a number of standard processing options and provide an opportunity for custom handling of service requests. The container will provide standard service discovery mechanisms for access to WSDL service descriptions on the fly or at startup.

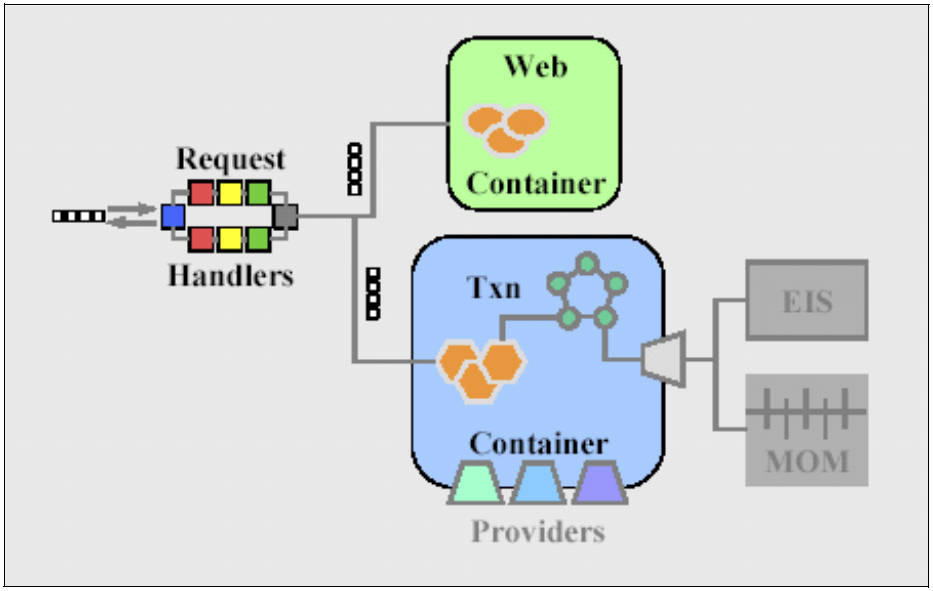


Figure 4-10 Service container technology

The value in the pipeline can be quite considerable. Both open-source Apache AXIS and J2EE 1.4 JAX-RPC specifications describe this approach for service request handling. Incoming message contents may be decrypted, digital signatures checked, clients authenticated, policy requirements validated, and SOAP header requests honored. On the reverse flow, outgoing messages will be decorated with policy requirements, logged for audit purposes, digitally signed, and encrypted as required.

The majority of J2EE vendors will be delivering service containers as extensions to their existing application server products. Much of the infrastructure has already been delivered by IBM in the WebSphere Application Server V5 products with proprietary extensions for wide ranging service access to existing assets through adapters and service gateways.

Enterprise Service Bus middleware

The idea behind de-coupling of clients from services has also prompted some thought leaders to propose the use of a “bus” concept for building middleware infrastructure in support of service oriented architectures. This concept allows systems architects and implementers to think about any-to-any connectivity for business service components.

The middleware product technology needed to provide such connectivity is logically an extension of technology already provided by many JMS and MOM

vendors. In addition to standard messaging, queuing, and protocol support, intelligent routing and broadcast facilities, and standard access API support, service discovery, service selection, and service request handling APIs would be needed.

Filters are available in the bus to apply custom or enterprise standard handling of service requests. Such filters may be used to validate and compress message contents, provide management functions such as metering and billing, or provide enrichment or suppression of service request information as required.

Using reliable messaging and standardized security measures, the service bus can provide assured delivery of documents between business processes and the scalability and robustness required by large enterprises. While the service bus will almost certainly be deployed within corporate firewalls at first, the potential for supporting B2B service interactions is real once reliable messaging and higher level standards are agreed and implemented.

Building a service infrastructure

As enterprises deploy service-oriented infrastructures, the starting point will often be a requirement to expose existing business functions to new clients (customer, supplier or employee portals), as well as standalone clients and other services acting as clients. J2EE application servers, like WebSphere, may be used to provide encapsulation of legacy business functions using WSDL service interfaces and to provide the service containers needed at run-time.

Over time, new services will be developed, off-the-shelf package services will be integrated into the infrastructure, and an internal registry of reusable assets will be defined. While much of this can be achieved with a single application server, enterprises will often have already deployed multiple platforms that must be integrated. At this point it may be advantageous to introduce a service bus to link the several service containers into a single manageable infrastructure (see Figure 4-11 on page 87).

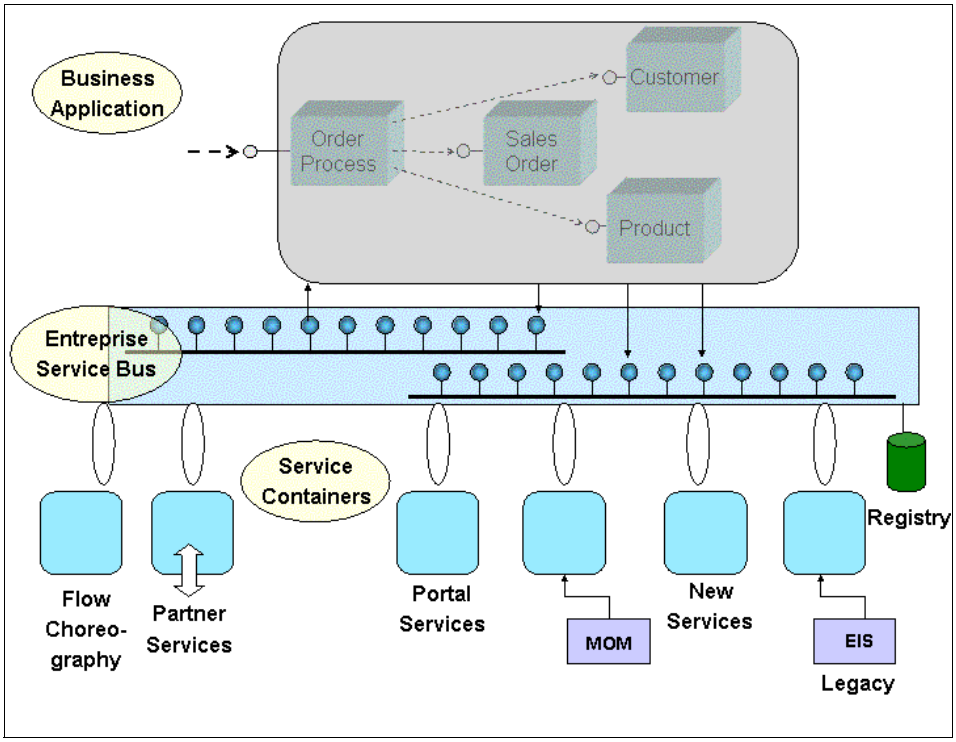


Figure 4-11 Enterprise Service Bus infrastructure

The Business Application is the solution which consumes services from one or more providers.

The Enterprise Service Bus provides a bridge between the implementations and the consuming applications, creating a logical view of sets of services which are available for use, invoked by a common interface and management architecture.

As standards evolve, such as BPEL, service flows will be deployed as points of integration that define higher level business functions. Automation of document transformation, enrichment, and logging of service messages will be integrated into the bus, but more sophisticated business workflow will require service flow state management and human interaction on an exceptional basis.

The service bus will also provide gateway connectivity and control for service requests coming from business partners and external users. The same gateway can serve to control internal access to business partner and Internet services, monitoring usage and applying enterprise policies to outbound requests as required.

Enterprise Service Bus summary

J2EE application server middleware products are evolving to implement open standard functionality that supports service description, discovery, deployment, and invocation in managed containers along with existing application components. Sophisticated tools are also being developed for easily modeling and mapping existing business application functions to service interfaces and a wide range of service adapters are becoming available as bridges to valuable legacy applications.

Much of the technology found in J2EE service containers can be reused and extended to implement the service bus concept. This integration and reuse of standards-based technologies can only be good news. Although the capabilities of the service bus and container infrastructure are focused on implementing the architectural service layer, they could be less expensive to deploy and operate over time as skills and other resources are also focused. All of this infrastructure is really just plumbing but it is also a sound investment in the foundation needed for the next architectural layer—the business process layer.

The goal for an SOA is a world-wide mesh of collaborating services, which are published and available for invocation on the Service Bus. The optimum implementation architecture for SOA is a component-based architecture. Many will be familiar with the concepts of process and entity component, and will understand the inherent stability and flexibility of this component architecture, which provides a one-to-one mapping between business entities and component implementations.

4.3.3 Adapters

A key element of any integration effort is to connect the various systems and activities that comprise daily operations. There are multiple applications, databases, Web interfaces and manual activities distributed throughout an enterprise that need to be linked together in order to be streamlined and optimized for better efficiency, cost savings, and flexibility. Companies not only have to be able to tie applications into their own system or infrastructure, but they must also allow adding and removing of external partners and suppliers from their business processes. Business connectors and adaptors allow companies to plug virtually any packaged applications into the business process without additional coding. For example, companies could purchase adaptors that connect to legacy host systems or SAP.

Adapters also allow integration of systems that are currently not enabled for Web Services.

The adapters are based on J2EE standards and provide common services to the connected application, such as:

- Common Client Interface (CCI). This provides a common application programming interface for the application programmer to communicate with the legacy application.
- A set of system level “contracts” that govern services such as connection pooling and management, transaction management, and security management.
- Resource adapter deployment and packaging. A resource adapter provider develops a set of Java interfaces/classes as part of its implementation of a resource adapter.

The J2EE connector architecture components are shown in Figure 4-12.

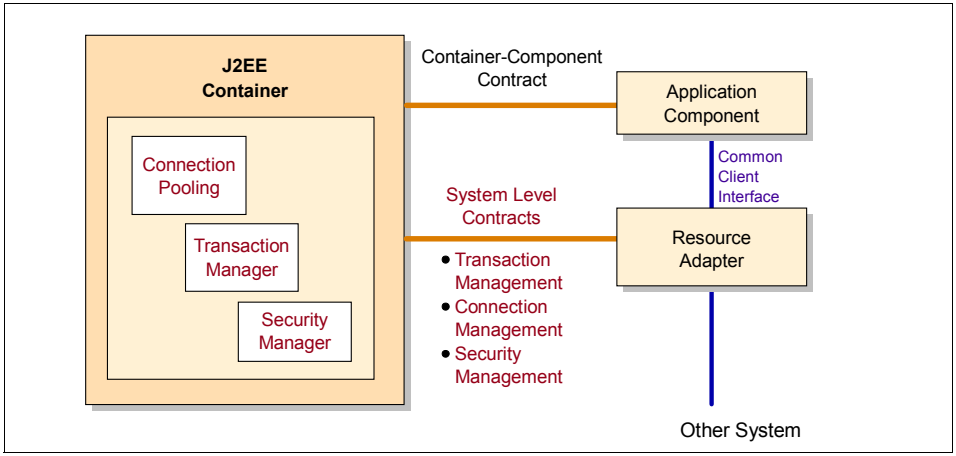


Figure 4-12 J2EE connector architecture components

As discussed in 4.3.2, “Enterprise Service Bus” on page 81, the ESB provides a business service oriented view of the IT applications and resources. Not all of these services are created from new applications or are enabled for Web Services. In fact, the ability to leverage the investment in existing legacy systems is a key feature of IBM’s approach to Service Oriented Architecture, without which the probability of success for any business process re-engineering effort would be very low in a realistic environment. Figure 4-13 on page 90 illustrates how adapters allow legacy systems to participate in the Enterprise Service Bus, thereby preserving and leveraging investment in such systems. They also reduce the risk and time of deployment of systems to support a business process re-engineering effort.

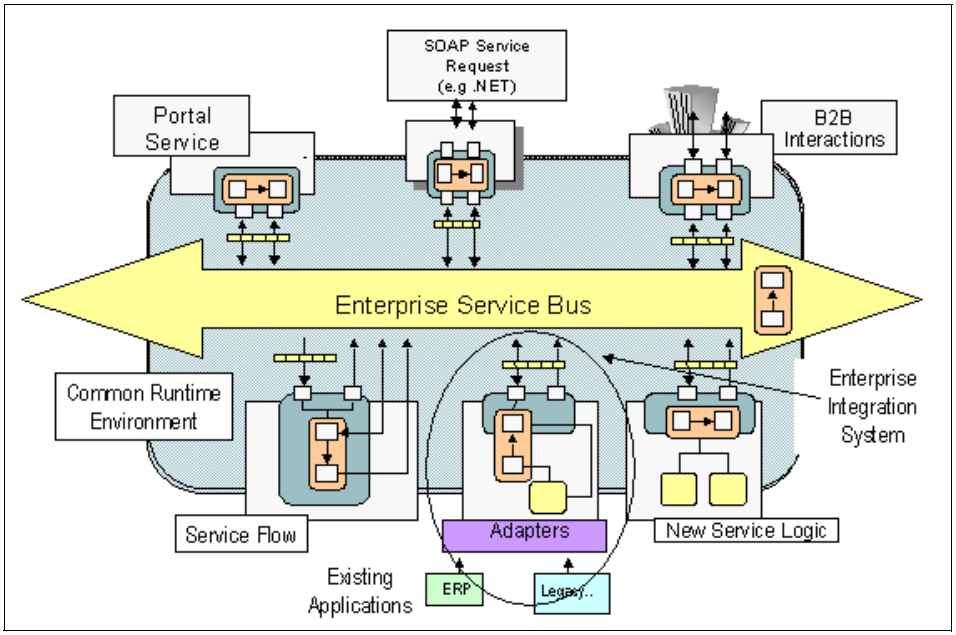


Figure 4-13 Service oriented architecture and adapters

As shown in Figure 4-13, there are multiple ways to integrate systems into the Enterprise Service Bus.

- Core functionality in legacy systems may be wrapped in Web Services in certain situations where code and licensing conditions permit. Many ISVs are increasingly creating Web Services that can be used to integrate their systems with others.
- In situations where it is not possible to directly enable an application for Web Services, adapters offer an alternative way of integration into the ESB.
- Resource adapter deployment and packaging. A resource adapter provider develops a set of Java interfaces/classes as part of its implementation of a resource adapter.

There are trade-offs to using Web Services.

The advantages are that Web Services enable a business to:

- Deliver new IT solutions faster and at lower cost by focusing their code development on core business, and use Web Services applications for non-core business programming.
- Protect their investment in IT legacy systems by using Web Services to wrap legacy software systems for integration with modern IT systems.

- Integrate their business processes with customers and partners at less cost. Web Services make this integration feasible by allowing businesses to share processes without sharing technology. With lower costs, even small business will be able to participate in B2B integration.
- Enter new markets and widen their customer base. Web Services listed in UDDI registries can be “discovered” and thus are “visible” to the entire Web community.

Some Web Services issues to consider are:

- Binding to Web Services dynamically requires that the contents of the UDDI registry be trusted. Currently, only private UDDI networks can provide such control over the contents.
- The SOAP server footprint is significant and the technology is relatively new, so adding the Web Service provider stack to existing enterprise systems can be a problem.

Adapters may be used with an Enterprise Integration System such as WBI or directly with WebSphere Application Server. Depending on the complexity of the connection topology between applications, one can choose IBM WBI Adapters or IBM WAS Adapters. As IBM WAS Adapters provide JCA adapters and JMS adapters, one can use IBM WAS Adapters in a simple topology of application connection.

As the number of connected legacy systems increases, the value of having an independent Enterprise Integration System (EIS) such as WebSphere Business Integration Server increases exponentially. WBI adapters provides the means for integrating legacy systems into the Enterprise Service Bus through WBI Server.

The next section provides information about IBM WBI Adapters and IBM WAS Adapters.

IBM WebSphere Business Integration Adapters

IBM WebSphere Business Integration Adapters allow customers to quickly and easily create integrated processes that exchange information between ERP, HR, CRM, and supply chain systems. WebSphere Business Integration Adapters provide several types of pre-built adapters and also provide a framework for development of custom adapters.

Adapter development tools

An integrated toolkit that provides a framework for development of custom adapters using both Java and C++.

Application adapters

Extract data and transaction information from both leading and industry-specific packaged applications. They enable access via Application Programming Interfaces where possible.

e-business adapters

Proven solutions for securely connecting over the firewall to customers' desktops, to trading partners' internal applications, and to online marketplaces and exchanges.

Mainframe adapters

Leverage a best-of-breed technology to access application data in OS/390 systems as well as provide connectivity approaches to AS/400.

Technology adapters

Provide various ways to access data, technologies, and protocols that enhance an integration infrastructure.

For detailed information on WebSphere Business Integration Adapters, visit the following Web site:
http://www.ibm.com/software/integration/mqfamily/adapter

IBM WebSphere adapters

IBM WebSphere adapters help customers link critical business applications into their e-business infrastructure, and also help them to share data between applications so that it is current and available throughout the affected enterprise systems. They eliminate the costs associated with the labor and errors of manual data re-entry between systems, or of unreliable batch file transfers on proprietary point-to-point connections.

J2EE Connector Architecture (JCA) compliant connectors connect applications with WebSphere Application Server using open standards.

4.3.4 B2B connections

The ability to connect to communities of trading partners allows for interactions with both suppliers and customers, thereby improving the management of partner relationships.

This section describes three ways to connect systems and applications between a company and trading partners or customers:

- IBM WebSphere Business Integration Connect
- Web Services

- Web Services Gateway

IBM WebSphere Business Integration Connect

IBM WebSphere Business Integration Connect:

- Enables complete community integration with trading partners and customers
- Is scalable from small to large communities
- Provides complete visibility of the trading partner community
- Enables rapid implementation of trading partner communities
- Leverages WebSphere MQ/JMS for WebSphere Business Integration connectivity
- Supports any data type: EDI, XML, Binary, Custom
- Supports multiple security trust models based on topology and partner requirements

IBM WebSphere Business Integration Connect also provides:

- Support for a wide range of industry-standard protocols including RosettaNet, AS2, EDI, and XML
- Support for trading partner interactions over transports such as HTTP(S), FTP, SMTP and JMS/WebSphere MQ
- Support for multiple Security Standards including 3rd party certificate authorities from Verisign and Thawte, SSL support, and Non-repudiation as required for full AS2 compliance

For detailed information on IBM WBI Connect, refer to the Web site at:

`http://www.ibm.com/software/integration/wbiconnect`

Web Services

Figure 4-14 shows a request-response Web Service invocation based on IBM WebSphere Application Server V5.0.2.

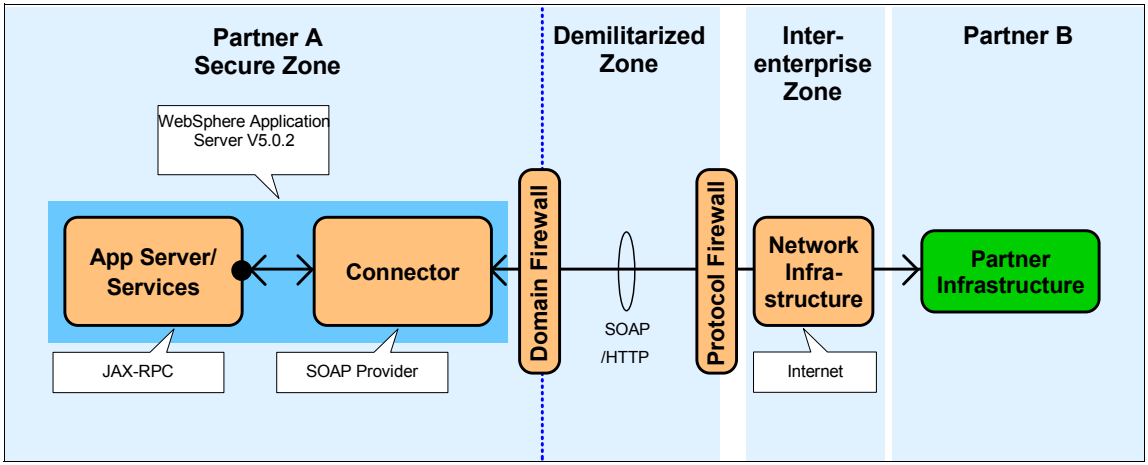


Figure 4-14 Web Services invocation

We use a connector in Partner A to model Web Services application integration. This emphasizes the use of a connector to convert the request and response into the common SOAP/HTTP protocol.

In this case, the source application uses the JAX-RPC API to initiate a request-response operation via the WebSphere V5.0.2 SOAP provider. Partner B receives the request from the source via its unspecified infrastructure.

Web Services Gateway

A Web Services Gateway can be seen as a kind of proxy that acts as an additional layer between a Web Services client and Web Services provider. The gateway enables a flexible way of calling Web Services located in an intranet from the Internet, as well as calling Internet Web Services from the intranet. Another function of the gateway is the possibility for protocol switching and security for Web Services calls.

Figure 4-15 shows a request-response Web Service invocation based on the Web Services Gateway packaged with IBM WebSphere Application Server Network Deployment V5.0.2 and the Connection.

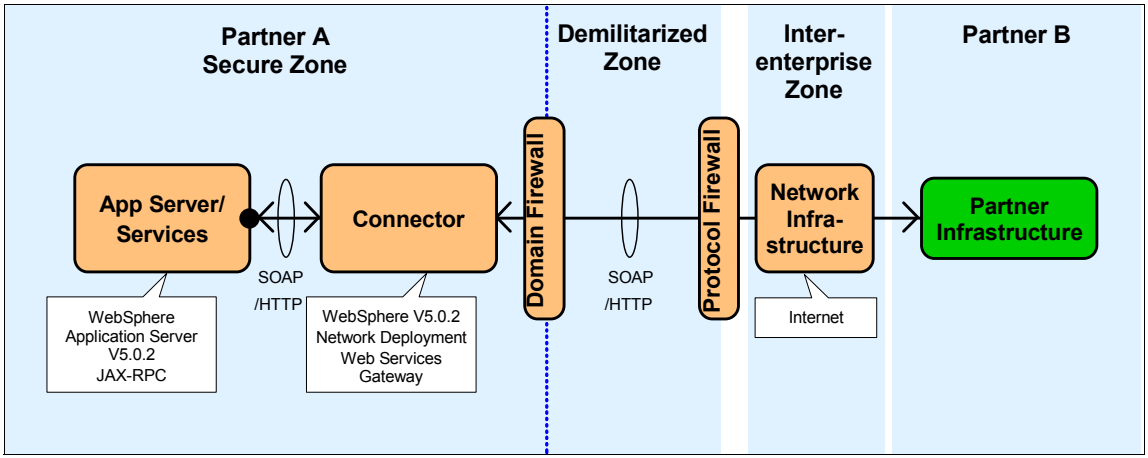


Figure 4-15 Web Services Gateway

The Web Services Gateway allows greater control over the point-to-point connection between the source application and a business partner's target application. The gateway provides access control and a common access point for external Web Services. It can also protect client applications from changes in the Web Services they access.

Web Services Gateway with protocol change

Figure 4-16 shows a request-response Web Service invocation based on the Web Services Gateway that provides a protocol change between Partner B and the target application in Partner A.

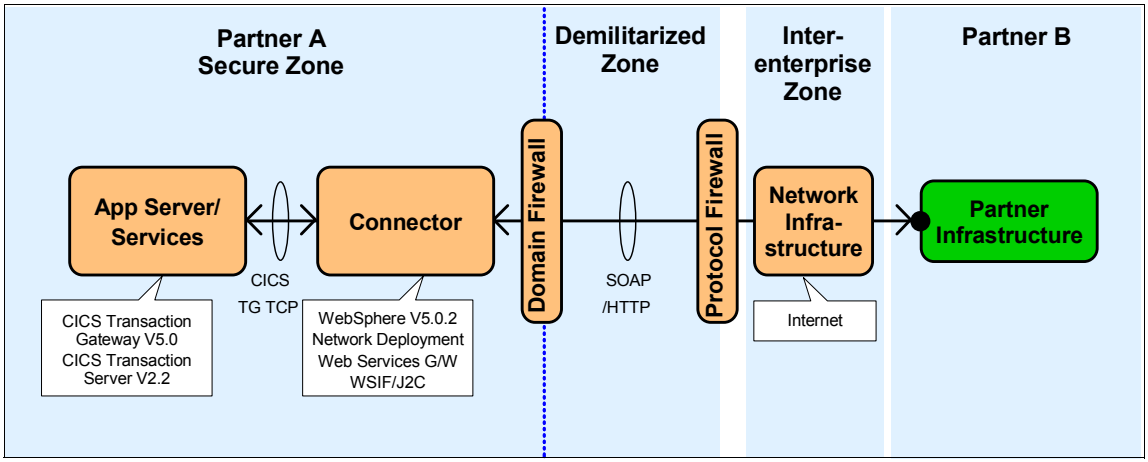


Figure 4-16 Web Services Gateway with protocol change

The gateway provides connector capabilities. A Web Service client application in Partner B invokes a CICS target application in Partner A using SOAP/HTTP. The gateway converts the SOAP/HTTP call to the CICS Transaction Gateway TCP protocol using the Web Services Invocation Framework and the CICS ECI J2EE Connector.

In addition to J2EE Connectors, the Web Services Gateway can be used to connect Web Service client applications with target applications that are accessed via JMS or RMI/IIOP.

4.3.5 Electronic Data Interchange (EDI) technology

Electronic Data Interchange (EDI) is widely accepted by companies all over the world as the way to electronically exchange business data. Business documents such as purchase orders, invoices, shipping notices, and price catalogs are exchanged between companies in a structured and computer-processable format.

Organizations are recognizing the value of many years of investments in EDI. Rather than replacing existing solutions, they are extending and evolving their EDI transactions. These existing EDI solutions are considered an integral part of a multi-modal B2B gateway or hub alongside XML, Web solutions, and portals. By integrating B2B and EDI technologies, event-driven or process-driven integration models can be supported using the existing EDI solution.

Advantages of EDI

The market is driving every business to act smarter and quicker and to be more visible. Much of this can be achieved using EDI. Even better, EDI can give companies more knowledge of their markets, since it opens up possibilities to collect and analyze information from the EDI transactions they are generating.

Among the most visible benefits of adopting EDI are:

- Reduction of data entry errors
- Reduced cycle time
- Minimization of paper use
- Improved relationships with business partners
- Easier sharing of information throughout the organization due to its electronic form
- Improved inventory management

EDI transmission

EDI is a *concept*. It does not define any techniques or point to any specified product or service. An EDI transmission can basically be divided into two logical parts: the message itself and the communication.

Message standards

Since the idea of EDI is to have a standardized message, a number of different standards have been developed and established over the years. The most commonly used message standards are:

- ANSI ASC X12 - US standard
- EDIFACT - Standard recommended by the United Nations, used mainly in Europe
- UNTDI - UK retail standard
- ODETTE - European automotive industry
- Others, such as HIPAA, VICS, VDA, UCS, and more

Communication

Transportation of the EDI file over a network can be accomplished in many ways. Any network and any protocol can be used as long as it fits the needs. Three common types of communication are:

- Value Added Network (VAN) communication

  Using a value added network for the transmission of files is traditionally seen as the most secure way of communication. Apart from doing pure communication, a VAN also provides value adds such as built-in security, restart and recovery facilities, archive capability, 24x7 availability, and notification of message arrivals.

- EDI over the Internet

  The initiative to move toward securely transmitted EDI messages over the Internet is known as EDI INT. Presently there are two main EDI INT initiatives, known as applicability statements, AS1 and AS2. They describe how current Internet standards can be used to achieve VAN functionality.

  - AS1 uses MIME (Multipurpose Internet Mail Extensions) and SMTP (Simple Mail Transfer Protocol).
  - AS2 uses MIME and HTTP (Hypertext Transfer Protocol) for process-to-process real-time EDI.

  The Internet solutions are often considered much cheaper than traditional VANs, but Internet solutions often leave it to the user to add functionality to achieve adequate security, reliability, and other features that are included in a VAN.

  IBM Business Exchange Services - Internet transfer is an example of Internet communication.

- Message queuing

  Message queueing (MQ) connects commercial systems in today's businesses. It provides assured, once-only delivery of data in any format.

  IBM WebSphere MQ is an example of this.

While the use of EDI technology is widespread, technology changes and evolution have resulted in the use of many types of B2B communication infrastructures. Besides the traditional VAN-based EDI communication, AS1 and AS2 Internet protocols are still tied more or less to traditional EDI communications. More recently, Web Services-based technologies also became available for use in the B2B area. While this technology is still maturing, it is clear that a flexible B2B solution should handle multiple communication techniques.

The Internet is widely perceived as being much less expensive than a VAN, but this is not necessarily the case. VANs generally provide valuable services, such as TPA management, service-level administration, security, and store-and-forward capability. The Internet requires that these elements are managed, which means the total costs are not always lower than a VAN.

Elements of an EDI solution

In addition to the obvious components of an EDI solution, such as application programs and systems, VANs, and trading partners, a complete and flexible solution should include the following important elements:

- Translators

  A universal problem in integration of applications is the conversion of shared data from one format to another. Common data fields—such as names, addresses, and numbers—often have different formats across disparate systems. The traditional approach to EDI implementation is to place the function that converts application data to the EDI standard directly into the business application. This approach is less effective because a separate program is required for each transaction as well as for each trading partner. In addition, it is difficult to keep up with new versions of standards because programs must be modified every time a trading partner adopts a newer standard or version of the standard.

  This approach has changed with the introduction of third-party translation software, also known as mappers. The translator is responsible for mapping application data to the specific EDI format and vice versa. This translation software is implemented in either a centralized engine or in an adapter. It must handle primary EDI standards as well as different and evolving versions of each standard.

- Batch enveloper/de-enveloper

  Typically, because VAN charges are based on each transaction sent, enterprises have been driven to find ways to reduce the number of transactions and to compress more information into each. Consequently, EDI messages are sent in large batches, which can then be grouped from, or split out to, several divisions or areas of an enterprise.

  Enveloping batch messages involves placing the EDI standard header and trailer around transactions in preparation for sending. When the envelope is complete, the package can then be sent to a trading partner through a VAN. Similarly, batch transactions must be de-enveloped when they are received from the VAN.

- Message router

  Once the EDI message is de-enveloped, it can be divided into function groups. Each function group may relate to a different division or area of the business. A mechanism is needed to sort messages destined for different groups and deliver them to the appropriate target applications. This means there is a requirement to fan in and fan out messages. Message transformation may also be required to get the message into the correct format for the target applications.

- Trading partner agreements (TPAs)

  A TPA is an agreement related to the exchange of information in electronic transactions. The term includes a particular agreed-upon standard for business documents, as well as communications and business protocols, the service-level agreement, and more. TPAs can also be extended to include business events. For example, if an event occurs in one organization that might affect processes in a second organization, the TPA can specify that the second organization be alerted to the event.

4.3.6 Common information and resource model

Processes typically act on data. To be able to make the kind of rapid changes described previously, it is critical that access to data is flexible and federated. This allows a new process to access the same underlying information sources and new information sources can be accessed by existing processes. This kind of flexibility requires the ability to dynamically federate new sources, create new views, and be able to invoke other services to access information.

For more details on information integration, see Chapter 5, “How to react in real time through seamless flow of information” on page 139.

4.4 Scenarios

This section describes two possible scenarios that address how one might implement an on demand Operating Environment to provide the capability to rapidly modify business processes.

The first scenario describes an environment where the majority of the work can be done by the business analyst. In the second scenario (4.4.2, “ABC Electronics scenario” on page 116) the IT department is much more involved in providing the IT infrastructure required to support the business analyst’s goals.

4.4.1 Manufacturing company re-engineering scenario

The case we describe here is based on a re-engineering exercise performed in a manufacturing company.

Business context

This case concerns a manufacturing company and its business of handling used equipment. This equipment is returned to the manufacturing plant following an end of lease or a repurchase. On arrival, the equipment is verified and stored until a customer order will re-use it. At that point, it must be re-configured, tested and shipped.

During the last 10 months this activity has undergone a lot of changes affecting the business process:

- There are more products to manage
- The customer demand has increased
- The internal organization changed, with a separation of the activities in three buildings, previously grouped in one location in one building
- Skilled resources were lost

Consequently the company’s performance within the industry has degraded. Management has decided to launch a re-engineering exercise, end to end, from the receipt of the customer order to the product shipment. The goal is obviously to meet business objectives and stabilize operations, but also to take the opportunity to optimize the use of resources.

After evaluating the current state of the business, the re-engineering team has targeted two specific areas to focus on:

- Information management
  - Communications between people in different buildings/locations has been difficult, and has resulted in a large increase in the use of internal e-mails and phones.
  - Materials tracking is largely a manual process with numerous paper forms.
  - Incomplete end-to-end process monitoring through different tracking tools for each component of the process.
  - Incomplete execution of disposition information has made it difficult to optimize materials usage.
- Priority management
  - Different rules are being applied depending on product types or parts to be purchased to complete an order.
  - Capacity conflicts, specifically for test cell resources.
  - Limited skill set of operators impacting production line capacity.

In this case, improved business integration can play a significant role in information structuring. At the same time, by providing management tools to closely monitor events, process improvement opportunities can be more easily identified and resources can be adjusted dynamically to meet specific spikes or troughs in the business.

Process description

The following figures provide a high-level illustration of the business process elements involved in this case.

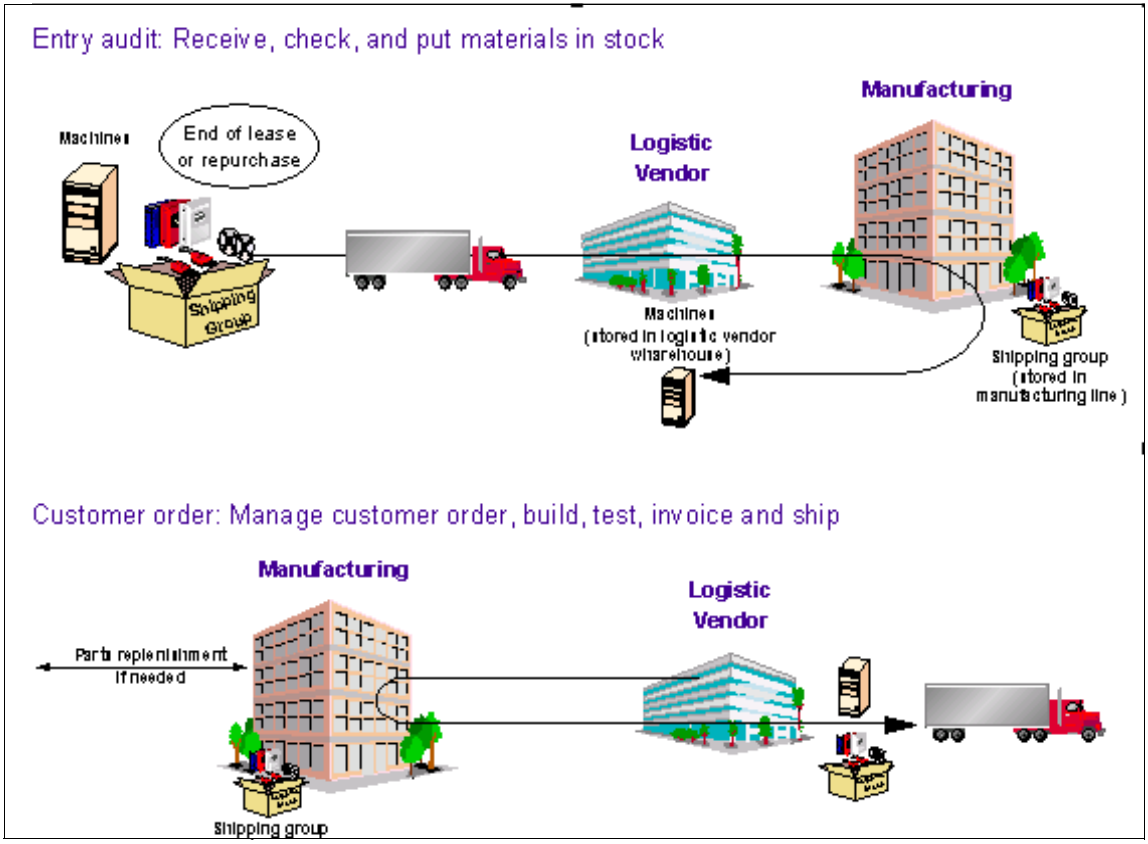


Figure 4-17 High-level view of business process

The second view, Figure 4-18 on page 103, shows the business process as a set of main functions.

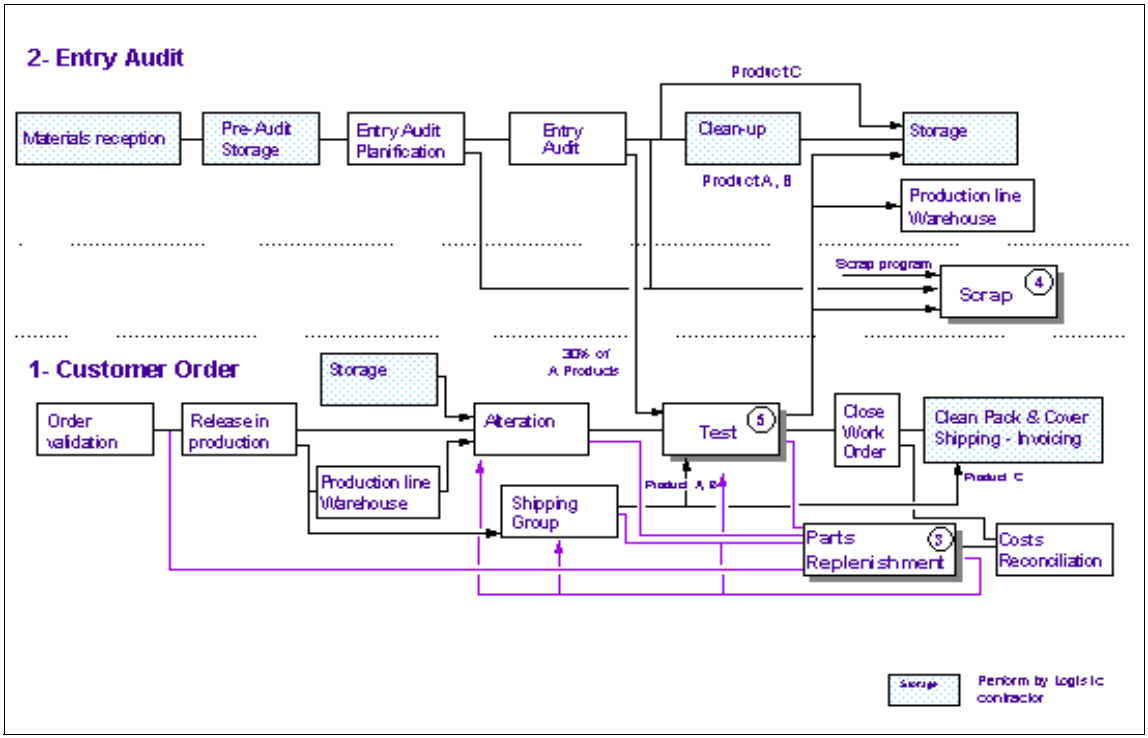


Figure 4-18 Functional overview

Customer order process

The customer order business process starts when orders arrive from a SAP system. This is the business process trigger. Orders are validated in terms of machine configuration and the supply in stock is checked before releasing the order to production. If the order is valid and supply is available in used stock, the production line receives the machines and features (memory, power supply, channel cards, and so on) and configures the order. Alteration performs hardware operations by adding and/or removing feature codes. Parts related to removed feature codes are stored in the production line warehouse.

In parallel, the production line warehouse produces the ship group which is dependent on the order configuration. Then, the machine is tested, the order is closed, and the machine is sent out to the logistics vendor that finalizes the order (clean, pack, and cover), and proceeds to the shipping and invoicing activities. The business process ends when the SAP sales system is updated after shipping and invoicing.

If the used inventory doesn't have all of the parts required to build the machine, or if alteration and shipping detect missing parts, or if test detects a defective

part, parts can be replenished from an external source according to price and replenishment time rules. When these parts arrive, the order processing is resumed at the step where it was stopped and can proceed to the end of the business process.

Entry audit process

In fact, the entry audit is the first step in this business process and consists of storing materials in the warehouses and recording them in the inventory system.

Materials are received in the logistics vendor warehouse and pre-stored until manufacturing initiates a request to check the configuration. At this time, if disposition instructions are available for the machine, meaning that only some feature codes are requested, the machine is delivered to test. The features with disposition instructions are tested, removed from the machine, and stored in the production line warehouse. This occurs for 30% of the machines. This allows the customer order process to respond very quickly to a “feature-only” order, especially since they have already been tested. In this case, the features need only be packed and they are ready to be shipped without performing any additional operations. The rest of the machine is sent out directly to scrap, along with the defective parts detected during test operations.

When materials configuration is checked, items are stored in two different locations:

- In the logistics vendor for machines (large dimension)
- In the production line warehouse for feature codes and shipping groups

Method to re-engineer the business processes

In summary, the manufacturer’s team adopted the following method to change their business processes:

1. Perform an as-is description of the business process with all the characteristics: who is doing what, when and with what inputs/outputs and with what mechanism.
2. Analyze, simplify, and re-engineer the process. Define all data required to quantify the business process parameters (volumes, organization structure, number of people, tasks/process timing, criteria, case percentage, and so on) and validate the new process with business integration simulation capability.
3. Get management support and manage changes.
4. Communicate with and educate people, implement the new process, automate through MQ Workflow, and measure the business process through a strong management system using business integration monitoring capability.

Redesign of the business process

We won't explain in this redbook the details of the business processes modeling, but we do show some functions of the WBI modeler tool that help create an accurate business process.

Prepare organization data

The organization structure can be easily created using the template provided in WBI workbench. It lists the organizations of employees involved in the business processes for customer order and parts replenishment. (See Figure 4-19).

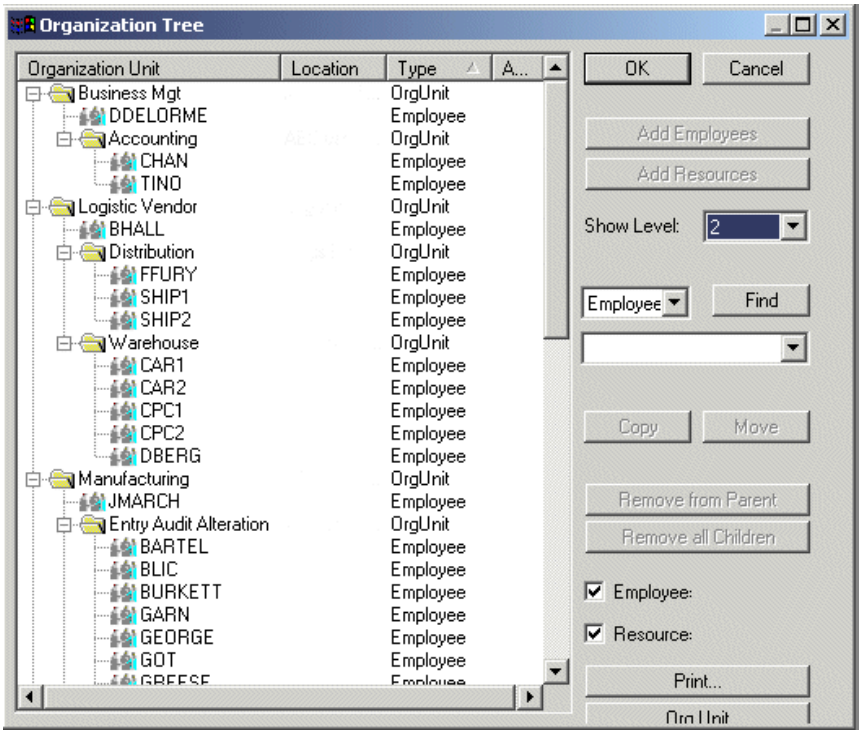


Figure 4-19 Organization data imported in WBI Workbench

The next step consists of creating the link between organizational units, primary units, managers, and locations. The result is represented in Figure 4-20 on page 106.

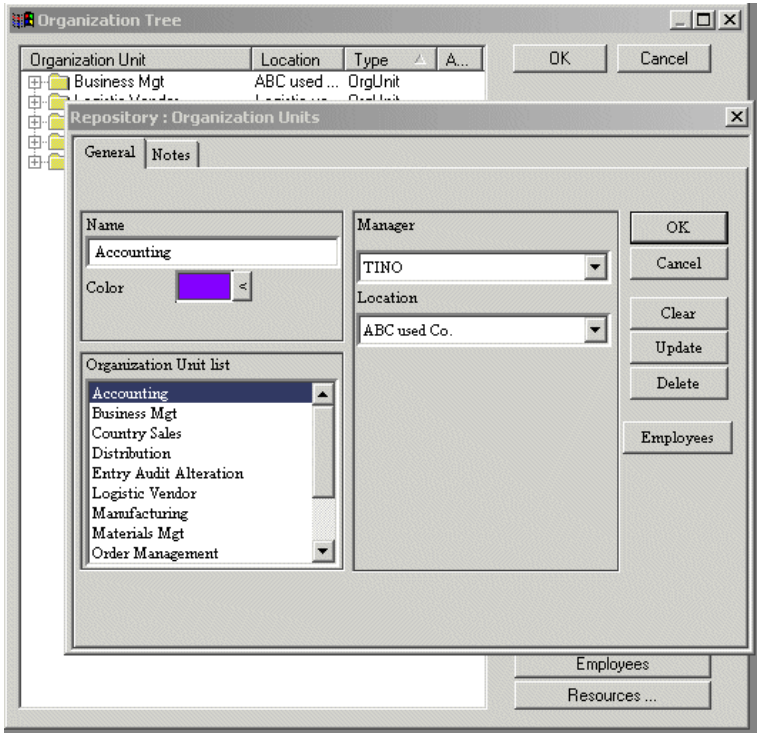


Figure 4-20 Final organization data

Represent the cost of the employees

This operation consists of entering the employee salaries/costs. This can be done by entering these costs in the role menu.

This enables calculating the cost of a business process.

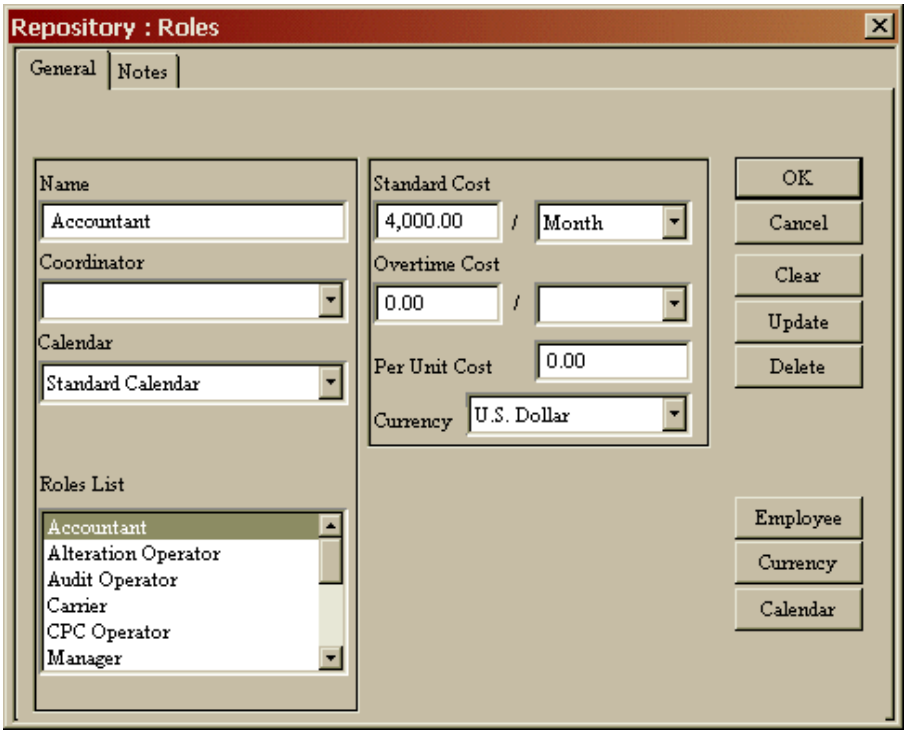


Figure 4-21 Set up role costs

Calendar information

After importing the organization structure, we need to set up the working time for the employees. Especially in a manufacturing company, this may include shifts.

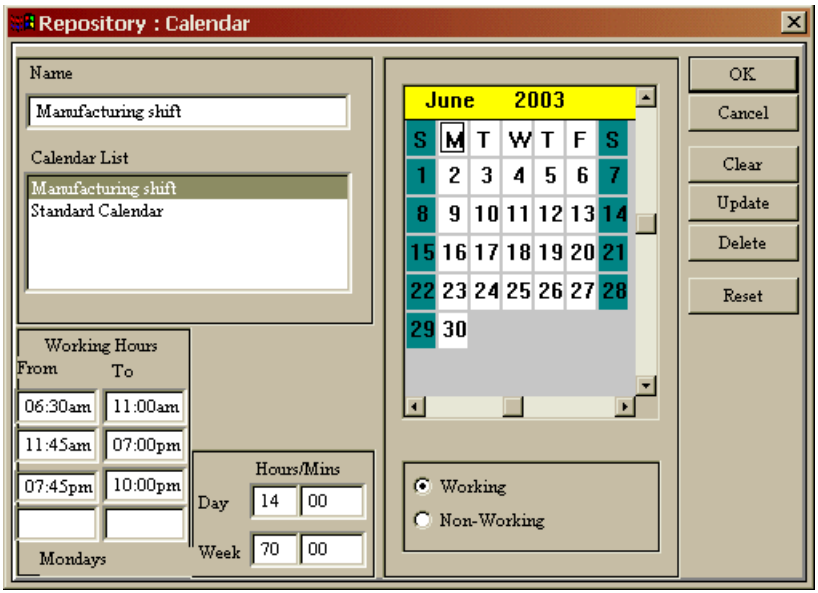


Figure 4-22 Creation of a calendar

Data structure set up

All of the information that is involved in the business process is represented in a data structure. The part of the data structure that is exchanged between two processes is called Phi, as the Greek symbol.

An IT process modeling tool, like MQ Workflow, reuses the data structures to represent the information that is exchanged between the application systems.

The data structure we have to handle for this business process is based on two objects:

- The customer order
- The parts replenishment order

These two objects will be modeled in the workbench as two data structures.

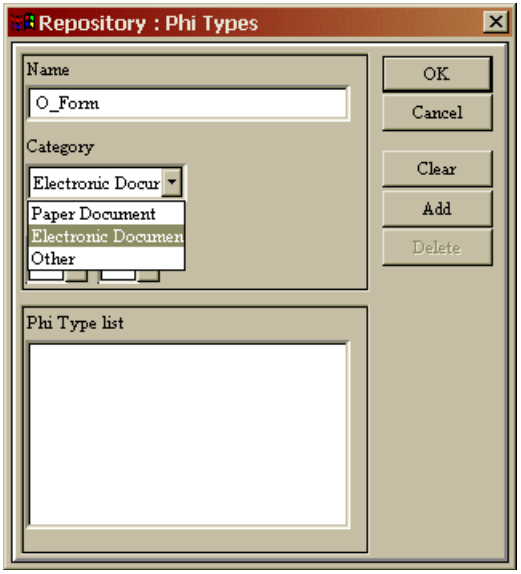


Figure 4-23 Set up the Phi Types

Building the logical flow

Based on the business description, we need to draw a logical flow before we model it in the WBI Workbench.

This part of the modeling process is critical. Modeling is a simplification of the real environment. The business analyst in charge of the modeling has the responsibility to find the most appropriate logic and associated key parameters to be able to simulate real behaviors without translating all possible cases. The quality of the modeling affects the quality of the recommendations provided to company management and, obviously, the quality of their decisions to tune the business process through monitoring.

Figure 4-24 on page 110 represents a snapshot of a logical flow.

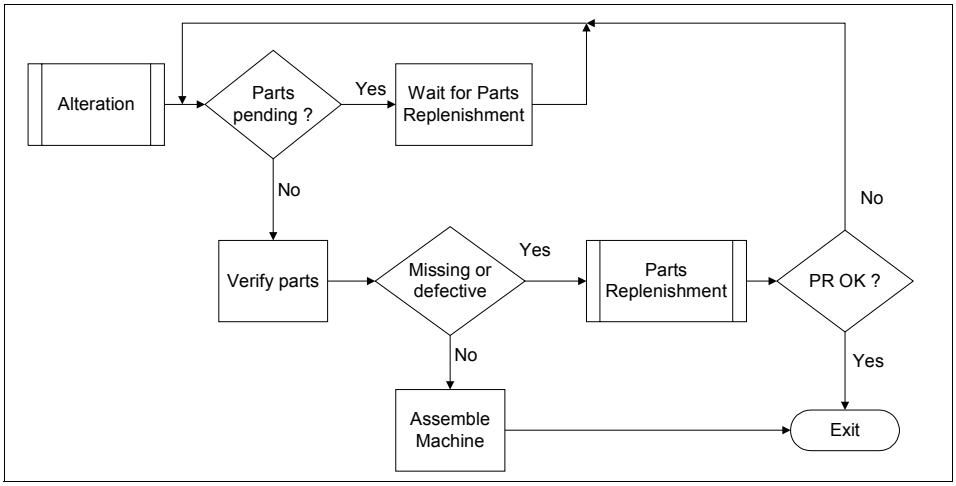


Figure 4-24 Alteration sub-process

Build the workbench model

Now all the elements to model the business process in the workbench are available:

- The business process flow with who is doing what and when
- The organization reflected in the workbench through the business units, employees, roles, costs, calendars, locations, and so on
- Business objects created through Phi's O_Form and PR_Form
- Logic flows that represent business process tasks and rules

Figure 4-25 on page 111 shows the beginning of the process.

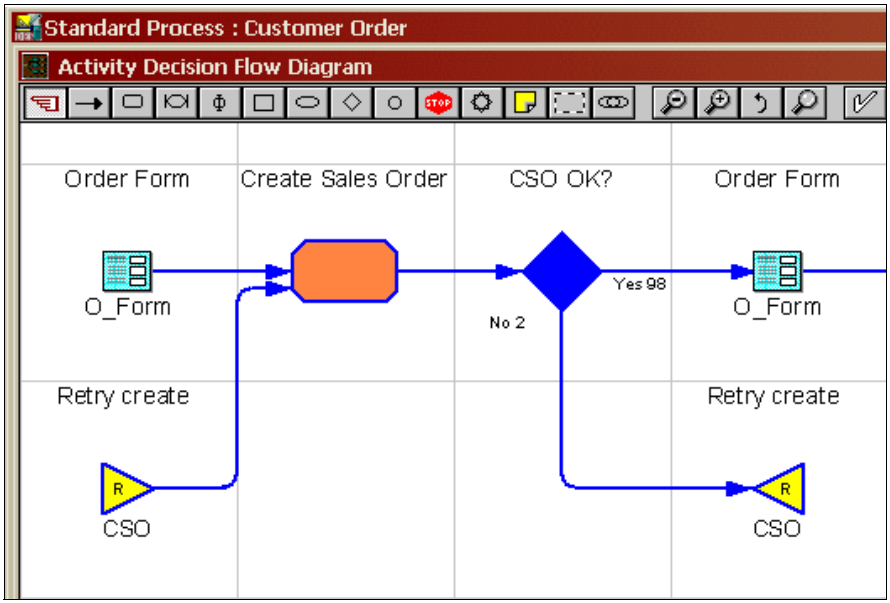


Figure 4-25 Customer order process beginning

Simulation of the business model

The simulation allows us to understand the business process behavior and is the basis for setting up business process measurements.

To validate a model, several steps must be completed to verify that its dynamic behavior is in phase with the real business process, based on the following characteristics:

- The model diagram properly takes into account the jobs in all the paths according to processes, subprocesses, rules, decisions, and quantities.
- The resources and the different calendars (task, resource, and simulation calendars) are properly managed by the simulation.
- Time values obtained are realistic: cycle time, queue time, and so on.

Each business analyst develops their own way to validate the model. But the three previous behavior characteristics will have to be checked anyway.

Several scenarios are tested to find the right job resources settings.

The scenarios analyze the job cycle average duration and the resource shortage time of the following main roles:

- PA: Product Analyst

- SCS: Supply Chain Specialist
- Carrier
- Alteration operator
- Test operator

Changing the number of resources for each role will impact the number of days to complete the customer order cycle duration. The results of the simulations are displayed in Figure 4-26.

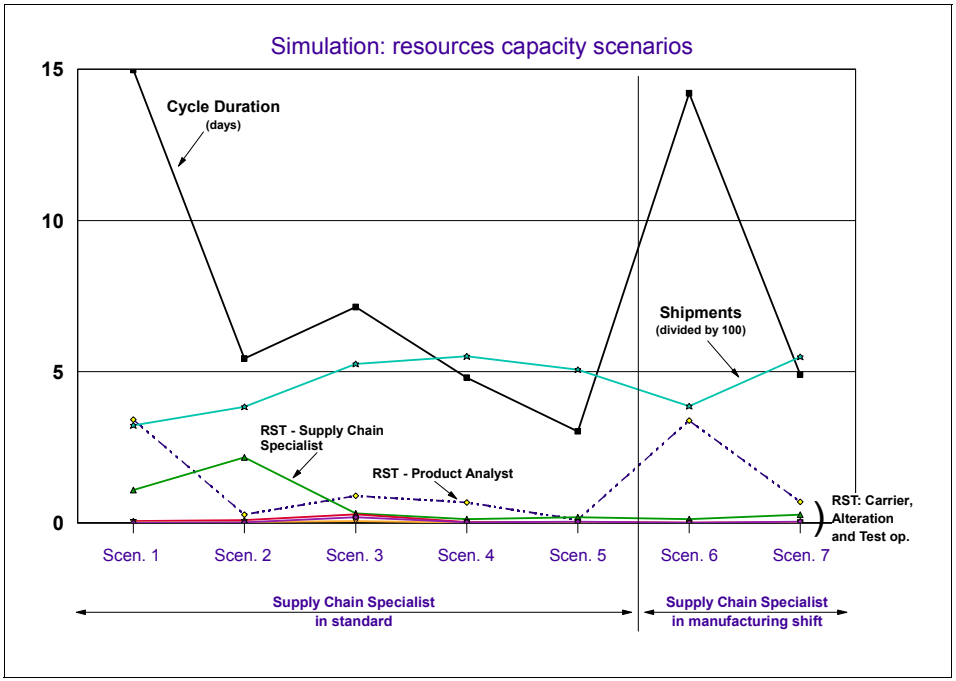


Figure 4-26 Simulations scenarios results

Building the user interface for the business workflow

Humans are critical actors in a business workflow and it is very important to explain to them the new processes. In this case, they will interact with the process through user interfaces that have been built with JSP technology.

The following figures show some of the interfaces developed for the new business process.

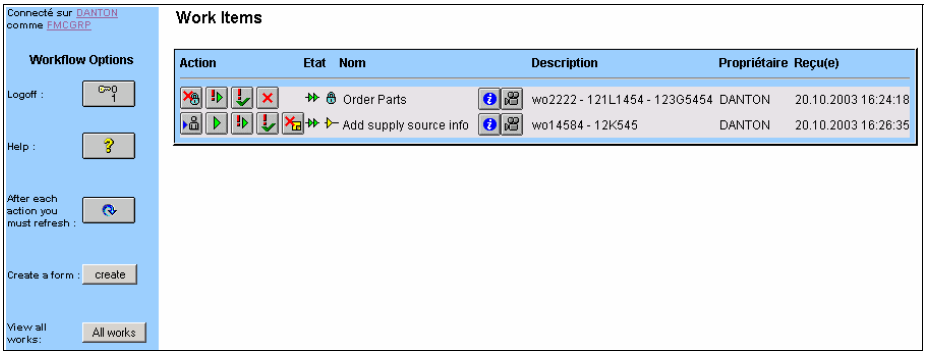


Figure 4-27 User application main page

The Menu is on the left side and the Task List on the right side.

Figure 4-28 represents a screen to request a new part for a server.

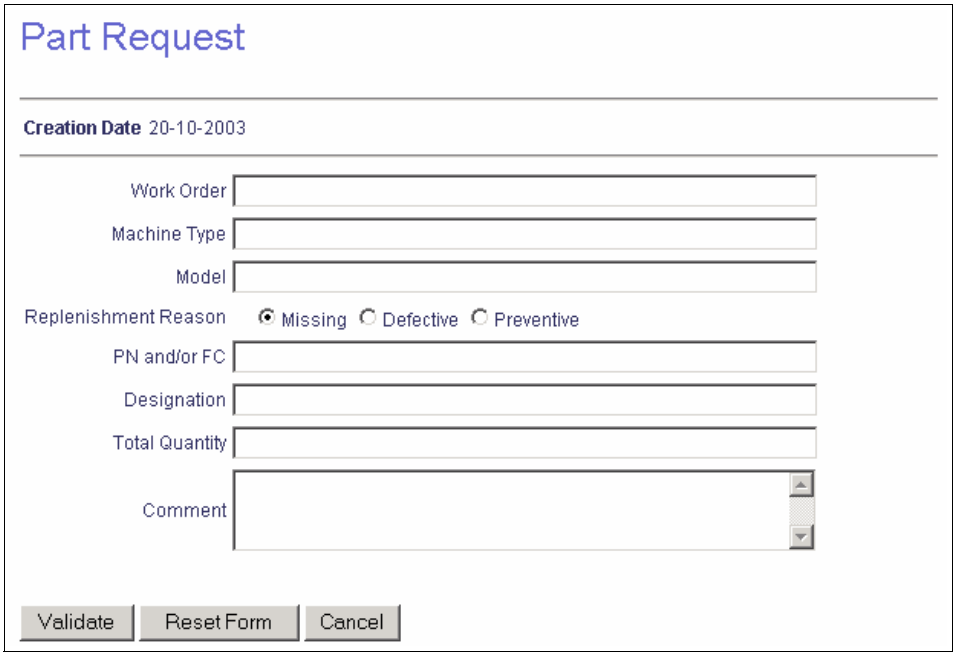


Figure 4-28 Part request screen

Description of the IT system

The technology infrastructure, as depicted in Figure 4-29, contains hardware and software platforms to support both the development and run-time environments.

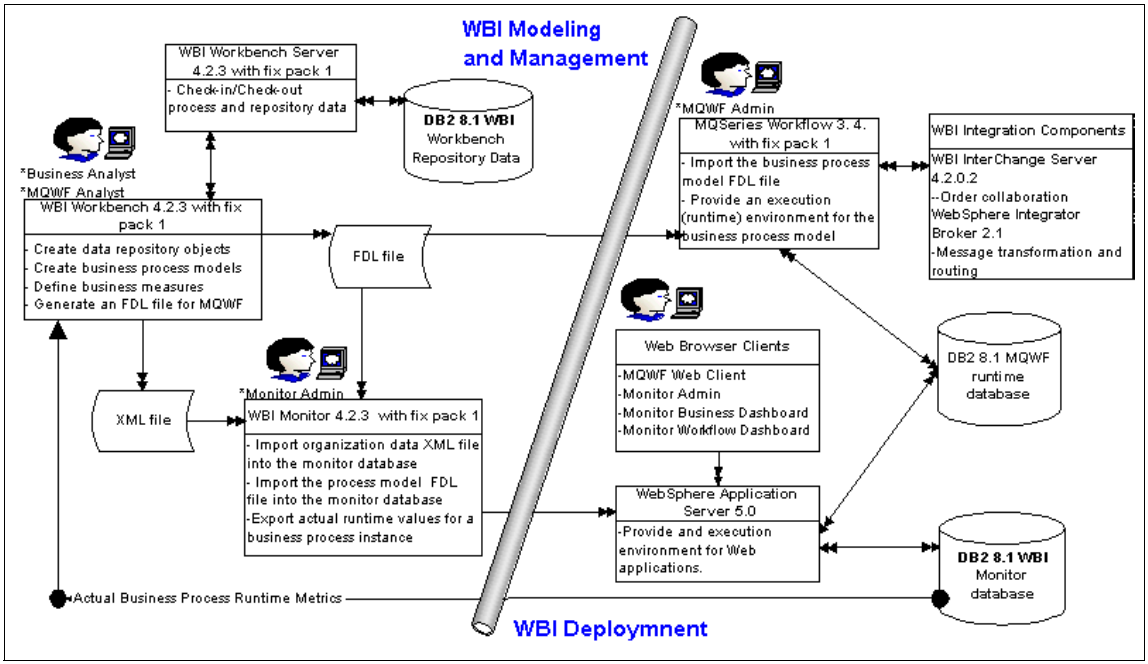


Figure 4-29 IT architecture diagram

Solution components

- WBI Workbench tool is used to model, simulate, and analyze business processes before deployment to a run-time container (for example, IBM WebSphere MQSeries® Workflow) or export to WBI Monitor.
- WBI Monitor enables an organization to monitor business processes and business performance measures, and generate real-time or historical reports using workflow and business dashboards.
- WebSphere Application Server is the IBM Java-based transactional application server platform that enables an organization to deploy Web applications and integrate back-end systems.
- WebSphere MQSeries Workflow is the IBM workflow product that supports the deployment and tracking of long-running business processes involving people and systems. In the context of a business integration management system, WebSphere MQSeries Workflow provides a run-time or deployment environment for business processes that are developed using the WBI Workbench. In addition, WebSphere MQSeries Workflow also creates actual

business measurement data that is used by the WBI Monitor as a source for analysis and management.

- WBI Interchange Server is the follow-on release of the IBM Crossworlds InterChange server product. It incorporates pre-built vertical business integration collaborations to facilitate the coordination of internal and external business processes.
- WBI Integrator Broker is the IBM message integration broker product that provides message transformation, routing and augmentation for dissimilar business systems.
- WebSphere MQSeries is the IBM message integration product that provides a distributed infrastructure for message creation, transport, and retrieval. In addition, the MQSeries system can be configured to assure once only message delivery and achieve other qualities of service.
- DB2 is the IBM relational database management system product that enables data storage, manipulation, analysis, federation, and business intelligence. In addition, DB2 supplies tools for application development and standard interfaces for application/mobile device integration.

Redesign of the IT system for the new business model

The manufacturing company already had an IT system based on MQSeries and DB2.

After the new business process is defined within WBI Workbench, an FDL file is exported to the run-time environment (IBM WebSphere MQSeries Workflow in our case). MQSeries Workflow builds its own run-time workflow from the FDL file and generates the MQSeries queues and the DB2 tables to run the business process.

In fact, very little technical skill is needed to put in place the new business process.

Benefits and summary

This scenario is an illustration of the re-engineering process described in “Alternative 1: Led by business analysts” on page 67. The WBI Workbench tool provides the flexibility to create the business process adapted to your particular needs, and to simulate, monitor, and modify it over time.

In this scenario, 60% of the time would likely be spent on WBI Workbench, 35% to define the User Interfaces, and 5% in WebSphere MQSeries Workflow to import the FDL file.

This scenario is typically suited for use by business analysts.

4.4.2 ABC Electronics scenario

Another scenario we can consider is a hypothetical retailing enterprise, ABC Electronics.

Business context

ABC Electronics is a retail electronics store that specializes in both consumer and business goods. Founded 30 years ago, the company has grown from a small local storefront to a large regional department store featuring televisions, computer equipment, stereo equipment, and household electronics. The company has a large wholesale business as well, supplying computer equipment, fax machines, copiers, and other business electronics to merchants throughout the region.

As shown in Figure 4-30, ABC Electronics sells to other retailers as well as directly to consumers, through various channels including phone, fax, and the Web, and through its storefront.

It has traditionally had close relationships with a set of suppliers, such as supplier A. These suppliers have over time treated ABC Electronics as a preferred customer, assuring them of product availability during peak demand times and also extending favorable credit options.

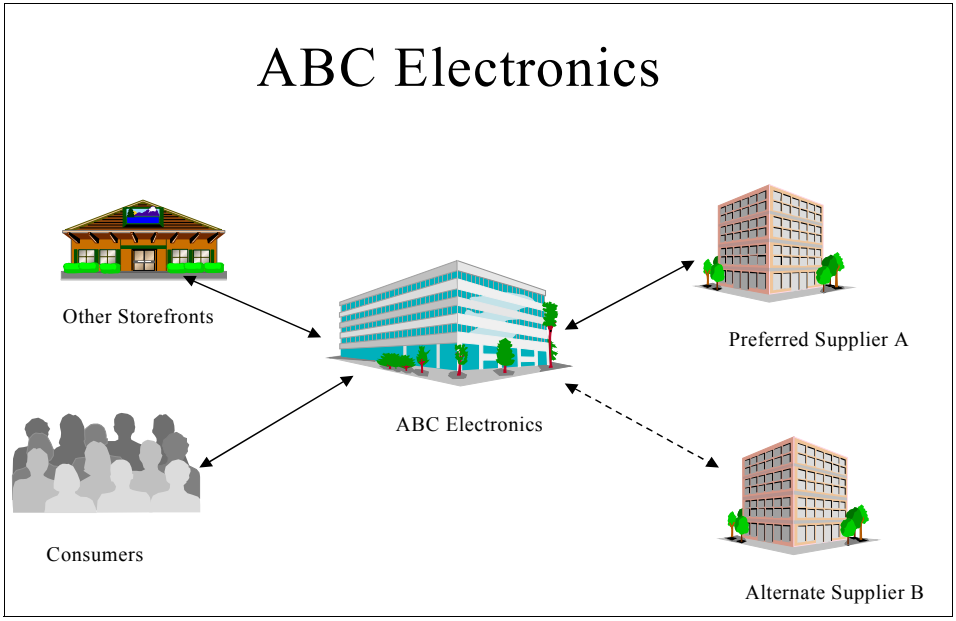


Figure 4-30 ABC Electronics: Business context

ABC has recently noticed that due to the continuous flow of new products from all over the globe, and availability of information about these products through the Web and other media, customers are requesting items that cannot be supplied by their traditional partners in a timely manner.

ABC would like to modify its stock replenishment process to allow other non-traditional suppliers to participate in case a particular customer request cannot be fulfilled by their traditional suppliers in a timely manner.

Current environment

ABC Electronics has two separate organizations, systems, and IT infrastructures for retail and wholesale operations of their business, with only paper-based, manual communications between them. The items sold are tallied at the end of the month by the retail ordering process and delivered to the wholesale organization by internal mail.

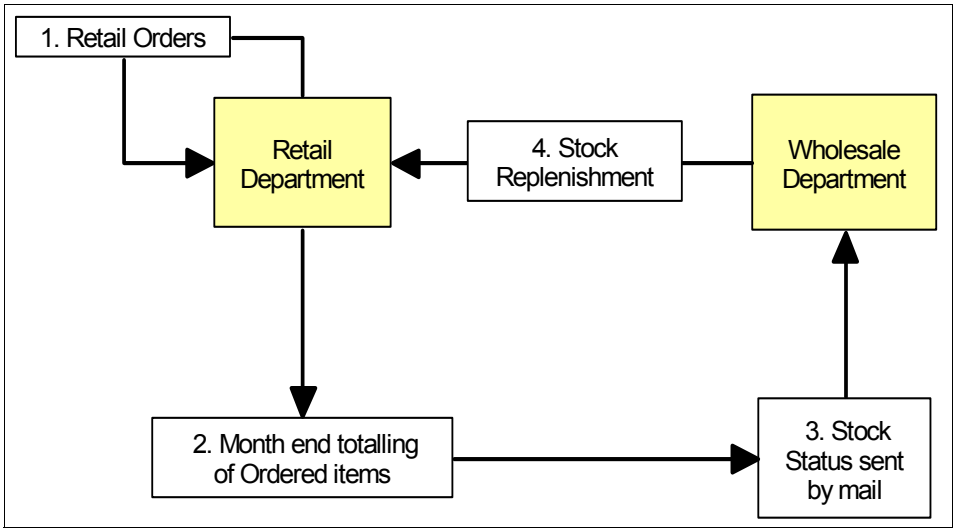


Figure 4-31 Replenishment process of current stock

This creates a lag in the inventory replenishment process and causes many out-of-stock situations.

ABC recently made some changes in their organization:

- ABC recently brought a new CIO on board. He has several years of experience in the retail industry and is interested in using technology more effectively.

- The retail organization of ABC has also recently hired a business analyst that is very interested in enhancing her productivity through use of any suitable tools.

The current IT environment to support the business process is shown in Figure 4-32.

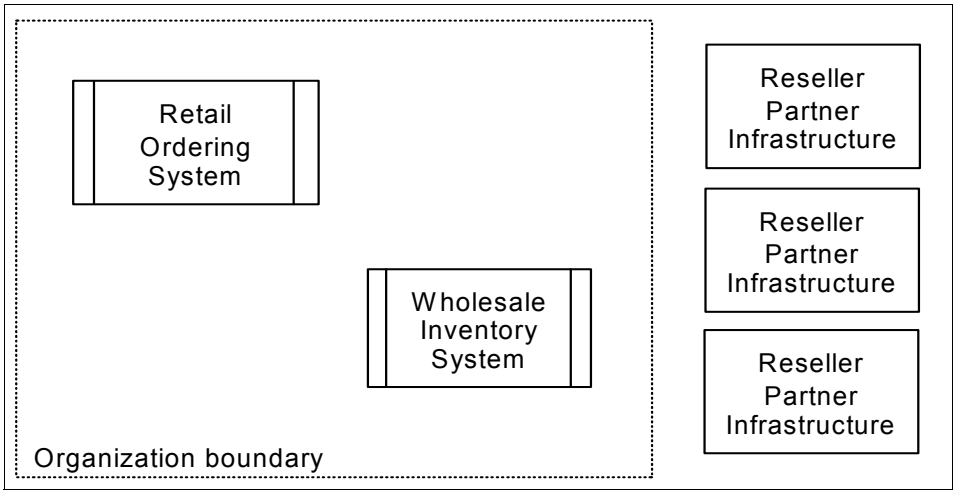


Figure 4-32 Current IT applications

It consists of:

- An existing wholesale inventory system. This important legacy system implements the core business processes of the wholesale department.
- An existing retail ordering system. This system is used by retail staff and has recently been upgraded to a self-service browser-based J2EE application.
- External resellers with their own, heterogeneous IT infrastructures.
- Limited integration between existing applications.
- ABC does not have a large IT programming staff. They have some Java skills, but their skills base is largely focused on the legacy systems they have in place.

A functional layout of the current systems is shown in Figure 4-33.

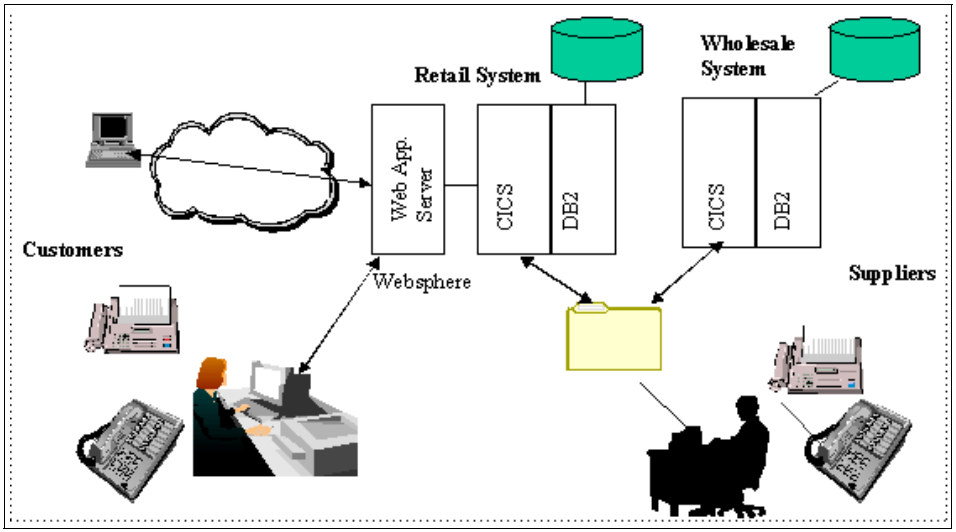


Figure 4-33 ABC - Current functional layout

Business objectives

One of ABC's primary goals is to minimize the loss of sales due to items being out of stock.

ABC recognizes that this will require closer integration of their retail and wholesale organizations so that customer requests can be monitored, and low levels of stock identified and replenished quickly.

They also recognize that they will need to communicate more effectively with non-traditional suppliers in situations where traditional suppliers are not able to meet their requests in a timely manner.

By integrating their retail ordering and wholesale inventory processes, along with their supplier systems, ABC Electronics plans to:

- Reduce costs by reducing the staff workload associated with placing stock replenishment orders with the wholesale department.
- Increase customer satisfaction by reducing latency between the retail ordering process and the wholesale inventory process, thereby decreasing the likelihood of an item being out-of-stock.

Technical objectives

In order to achieve their business objectives, ABC would require an IT infrastructure that would allow the desired level of integration between their internal systems and supplier systems.

Functional requirements

As a first step, ABC would like to integrate their internal retail and wholesale systems. They would like to do so in a manner that allows them to set up a foundation that is based on open standards, and has the flexibility to adapt to future business needs in a rapid manner. At a later stage they would like to leverage this foundation to integrate ABC's systems with other supplier's systems. Figure 4-34 shows the functional requirements.

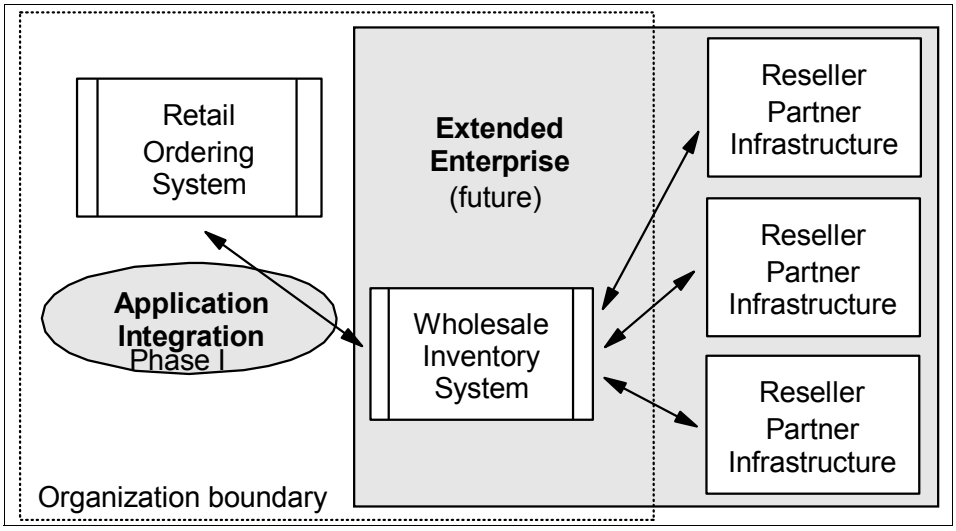


Figure 4-34 Functional requirements

Non-functional requirements

From a non-functional perspective, ABC Electronics requires that all solutions provide a standard Quality of Service (QoS) set. The following specific criteria must be met:

- Autonomic
  - Solutions provide suitable system management tools, procedures, and logs.
- Availability
  - Solutions meet both the defined unplanned and planned downtime requirements.
  - Meaningful messages are provided to system users during downtime.
- Federation
  - The responsibilities of the stakeholders are clearly defined and agreed to by all parties.

- Performance
  - Solutions meet the defined throughput and response times.
  - Solutions scale to provide for future growth.
- Security
  - Sensitive systems and data are protected from unauthorized access.
  - Non-repudiation of the end user for all commercial transactions is provided.
- Standards compliance
  - Appropriate standards are identified and applied.
  - Compatibility with existing internal systems and partners is considered.

It is beyond the scope of this redbook to define such requirements in real, measurable terms for our sample scenarios. Of course one would do so in an actual implementation to ensure that the delivered solution meets the demands of the organization.

Solution approach

The solution approach includes two aspects: redesign of the business process and IT system redesign to support it.

Redesign of the business model

ABC has made a decision to have the business process redesign drive the system redesign.

As mentioned earlier, ABC has hired a new business analyst. She is keen to use automated tools for describing business processes, modeling them as well as running various simulations to determine the optimal process. ABC purchases a copy of the WBI Modeler product for the business analyst to use.

It does not take the business analyst long to learn the basics of the WBI Modeler Workbench, which has an intuitive, easy to use graphical user interface. The business analyst then goes through the following sequence of steps with the WBI Modeler to determine the optimal business process for stock replenishment:

- Modeling the business process as it exists today. This includes manual, people-oriented tasks as well as applications. The business analyst assigns “costs” to the various tasks associated with the process today, in terms of monetary costs and time taken for each of them.
- Modelling the to-be scenarios. The business analyst develops several scenarios of the proposed process and ways to accomplish the business objective of lowering stock replenishment times.

- Simulating the business process to come up with the optimal model. This capability of WBI Modeler allows the business analyst to model and project various scenarios to come up with an optimal business process quickly. The output of the WBI Modeler for the selected business process model is shown in Figure 4-35.

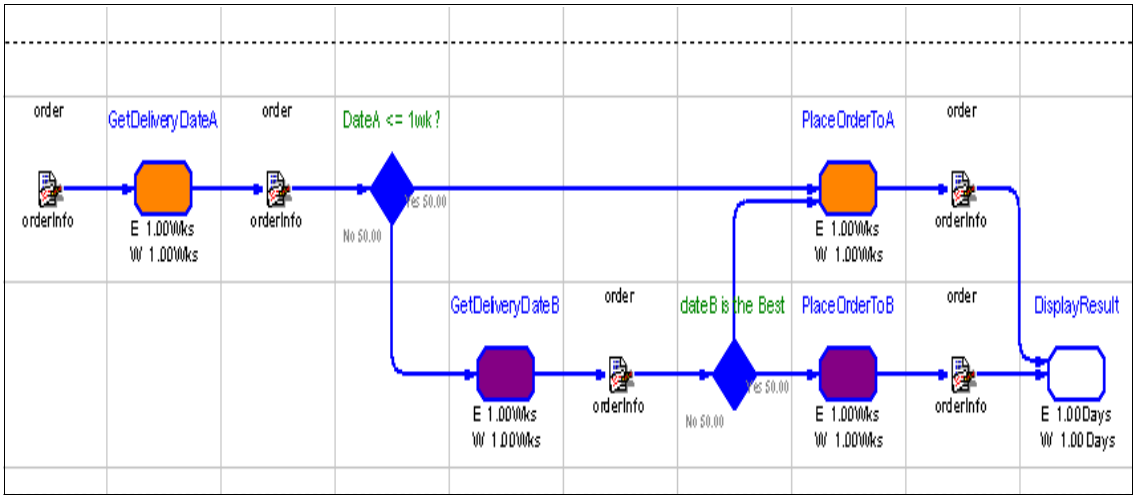


Figure 4-35 The redesigned business process

The business process is redesigned to reflect the following steps:

- The preferred supplier (A) is queried to obtain a date for replenishing the stock item in question.
- If the date obtained from the preferred supplier (A) is less than one week, an order for the item is placed with them.
- If the date obtained from the preferred supplier (A) is longer than one week, an alternate supplier (B) is queried to see if the item is available and the availability date is obtained.
  - The dates obtained from suppliers A and B are compared.
  - If B is able to replenish the item at an earlier date, the order is placed with them.
  - Otherwise the order is placed with A.
- The depleted inventory is replenished.

As a result of this analysis, ABC is able to establish business performance measurements associated with this redesign. Examples are:

- Actual time taken to replenish stock.

- Volume of sales lost due to low stock conditions.

Redesigning the IT systems to reflect the new business process

Once the business process has been redesigned, the WBI Modeler provides some options for how it will be deployed on the IT infrastructure.

The business analyst could have directly deployed it to the IT infrastructure, with minimal need for additional programming. However, this currently requires a WBI MQ Workflow infrastructure. Since ABC does not currently use MQ Series and WBI MQ Workflow, this could represent a significant investment in IT resources (hardware, software, and skills).

However, ABC has adopted an IT policy of deploying systems based on open standards and as a result has built up some skills in the J2EE environment and Java programming. They also have some experience with WebSphere Application Server which has been used in the past to Web-enable their retail system.

As a result, the best decision for them with this set of circumstances appears to be to team with their IT department to implement the redesigned business process on a WebSphere-based system.

The business analyst uses the UML export capability of the WBI Modeler to export the UML files representing the redesigned process for the IT department to use in their system redesign efforts. The exported UML for an example use cases is shown in Figure 4-36, in a format that is useful to the IT analysts.

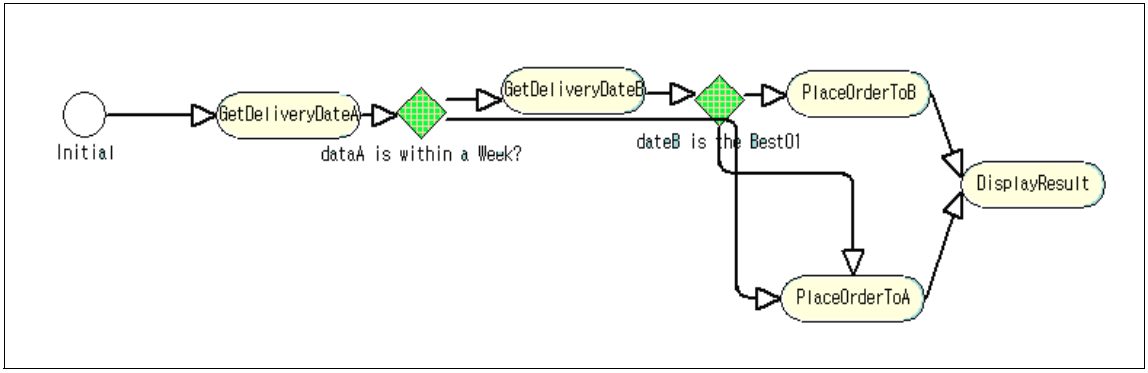


Figure 4-36 UML example

Functional updates

The UML is taken by the IT analysts and architects at ABC and used to architect the functional requirements for the redesigned system. They would typically use an environment such as the Rational Rose environment to import the UML from

the WBI Modeler and the Rational XDE environment to model the IT environment and create an architecture for the developers.

The first stage could represent ABC integrating the retail system and the wholesale system. The primary goal is to integrate the internal retail ordering system with the internal wholesale system to update inventory as replenishment orders are placed. The wholesale group will then be able to monitor inventory levels and deliver replacement parts as needed. There is no business requirement for confirmation when the inventory is updated, but there is a requirement for an audit trail.

For this stage there are two primary actors:

- The retail system
- The wholesale system

We can also identify a use case:

- Update inventory

The actors and use case are described in the following three tables.

Table 4-1 Retail System Actor

Actor name	Retail system
Brief description	The retail system implements the retail ordering business process
Status	Primary
Relationships	001 Update Inventory, 002 Get Delivery Date
Associations to use cases	

Table 4-2 Wholesale System Actor

Actor name	Wholesale system
Brief description	The wholesale system implements the wholesale inventory management business process
Status	Primary
Relationships	
Associations to use cases	

Table 4-3 Inventory System update use case

Use case name	001 Update Inventory.
Subject area	Wholesale ordering.
Business event	An item sold by the retail division needs to be replaced from the wholesale inventory.
Actors	Retail system, Inventory system.
Use case overview	The retail system places a replenishment order for a sold part with the inventory system.
Preconditions	The retail system supplies a part number for the item to be ordered.
Termination outcome 1	The wholesale inventory system logs the order to the audit trail and schedules delivery of the required part.
Notes	

The use case model is shown in Figure 4-37.

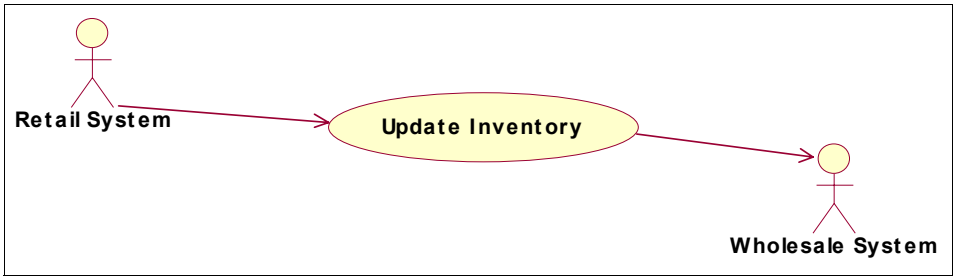


Figure 4-37 Use case model for updating inventory

In the next level of integration, ABC Electronics can further integrate their internal retail and wholesale systems. This will allow a real-time notification for out of stock situations with a delivery date indicating when the order can be filled. To improve customer service, the retail department wants to be able to quickly provide their customers with an expected delivery date for items that are not in stock with the retail department.

For this capability, the actors are the same as those identified earlier. There is an additional use case, as described in the next table.

Table 4-4 Use case to get allow notification of delivery date

Use case name	002 Get delivery date.
Subject area	Wholesale ordering.
Business event	A delivery date for an item out of stock with the retail division needs to be obtained from the wholesale system.
Actors	Retail system, Inventory system.
Use case overview	The retail system requests a delivery date for an out of stock part from the inventory system.
Preconditions	The retail system supplies a part number for the out of stock item.
Termination outcome 1	The wholesale inventory system logs the request to the audit trail and returns the expected delivery date of the required part to the retail system.
Notes	

The use case model is shown in Figure 4-38.

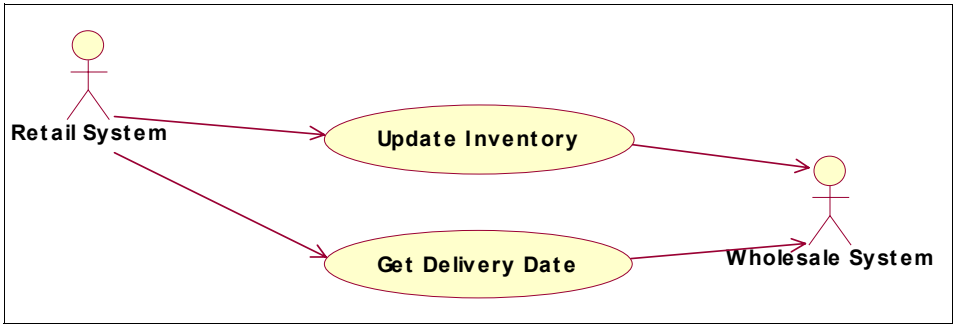


Figure 4-38 Use case model for updating inventory and getting delivery date

By leveraging use case models, the IT architect or IT developer models the business workflow. Figure 4-39 shows ABC Electronic's business workflow using WSAD IE.

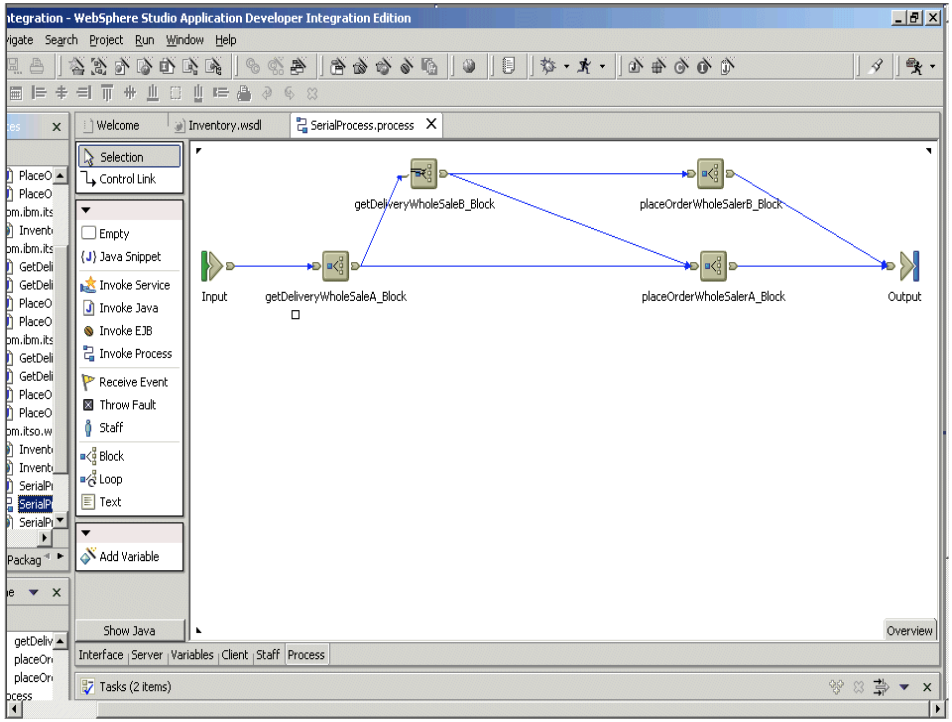


Figure 4-39 Business workflow using WSAD IE

Figure 4-40 shows a conceptual IT architecture considered by ABC Electronics to support their redesigned business process.

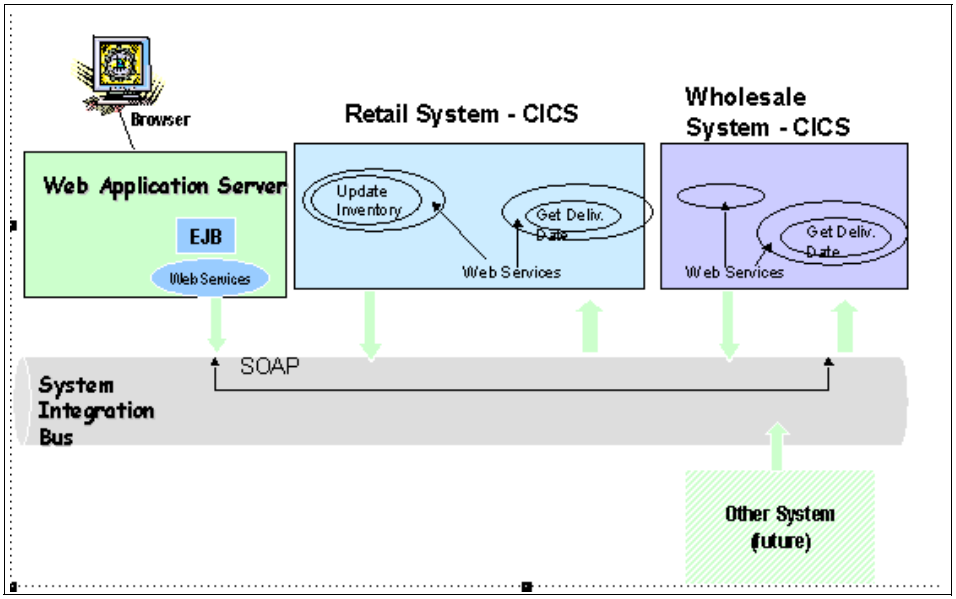


Figure 4-40 Conceptual integration architecture

Infrastructure updates

ABC has WebSphere Application Server installed, they have developed some J2EE and Java programming skills, and they are familiar with the WebSphere Application Developer Studio - Integration Edition.

The architects and developers in ABC's IT function are able to use the Eclipse environment for a common look and feel around their analysis and development efforts.

At this point the IT department needs to determine an IT architecture that will be implemented to support the redesigned business process. In order to do this there are several guidelines the IT department has established, such as:

- The Web is a preferred channel for interaction with users, both internal and external.
- Leverage investment in existing applications to minimize risk and lower costs of development.
- Develop a standards-based infrastructure that is flexible and can communicate with a wide variety of systems, both internal and external, such as suppliers, customers, and partners.

Based on these objectives, here are several alternatives the IT department can consider to integrate their retail and wholesale systems:

- Using native CICS communication between the retail and the wholesale systems. This approach would preserve investment in their existing systems. Retail customers and service representatives could access the system using a Web browser. However, users of the wholesale system would continue to use the traditional green screen interface. It would also not lay the foundation for an open infrastructure. This is contrary to the guidelines identified previously.
- Using traditional WebSphere CICS connectors to the two systems and developing the re-engineered business process on this infrastructure. This approach would meet the guidelines to leverage investment in existing systems, and provide a Web browser based interface to users. However, it may not provide the most flexibility to interoperate with other systems in the future.
- Integration based on a Services Oriented Architecture (SOA). The elements of this architecture would be:
  - Leverage core legacy systems and communications between them.
  - Transactions from these systems are enveloped into Web Services. These Web Services are exposed via an Enterprise Service Bus to be choreographed into the business process.
  - These service components are aggregated at a coarse-grained level to support new business processes, and promote integration and interoperability with other systems.
  - The service components themselves may be assembled and aggregated to promote reuse and provide rapid time to value for any new products or processes being considered. Typical guidelines are that no more than five components (representing transactions) should be aggregated to expose a “service.”

A high level architectural layout is shown in Figure 4-41.

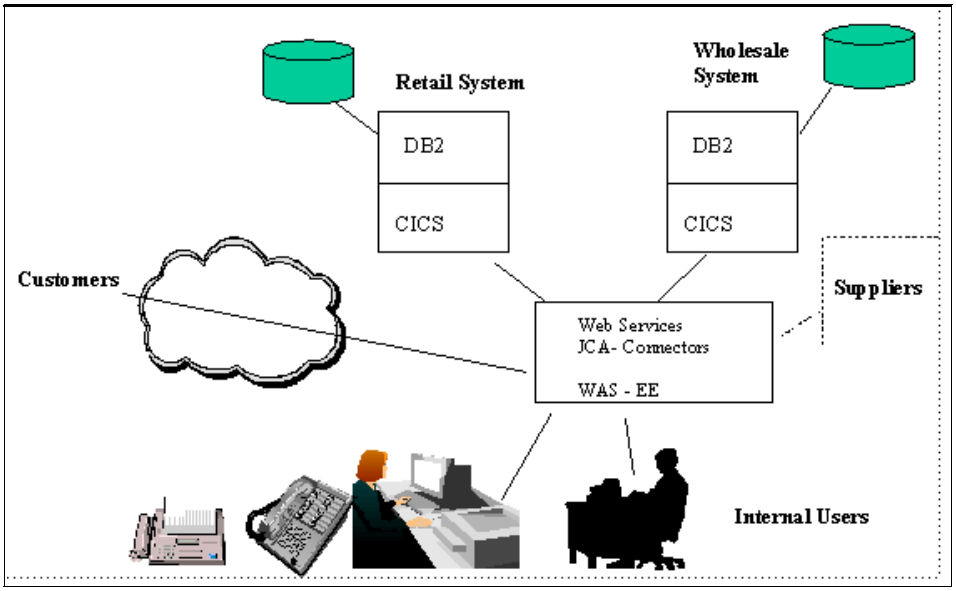


Figure 4-41 Proposed functional layout

Benefits and summary

In this simple scenario, we saw how a hypothetical retail enterprise re-engineered its business processes rapidly. Some of the key benefits obtained by them include:

- Ability to rapidly model and simulate business processes to determine the optimal process for their business needs and to set up business performance metrics, such as sales lost due to out of stock situations, time to replenish inventory, and so on.
- Ability to translate these business processes onto the IT infrastructure rapidly through the use of tools and a services-oriented architecture, which is flexible and extensible, based on open standards.
- ABC was able to leverage its investment in existing systems. The approach discussed in this section was an evolutionary approach, which did not involve “rip and replace.”
- This approach allowed ABC Electronics to build a foundation for future capabilities, enabling them quickly to support new products, relationships, and markets.

4.5 Product positioning

Several products were mentioned in the previous scenarios. We have summarized some of the differentiating factors among the various business process modeling tools and business process execution engines in the next two tables.

Table 4-5 Business Process Modeling Tools

WBI Modeling	Rational / XDE	WSAD IE
Produce FDL flows	Import UML flows/objects	Produce FDML choreography
Produce UML flows/objects	Produce UML/XMI flows	IT oriented
Business oriented	IT oriented	

Table 4-6 Business Process Execution Engines

MQ Workflow	WebSphere InterChange Server (WICS)	WAS V5 Enterprise Edition
- Execute FDL flows - Supports user interactions - Support long running flows	- Execute collaborations - Can import BPEL modeled flows and UML/XMI activity diagrams - Targeted to applications integration - Rich set of connectors - No support for user interactions	- Execute service choreography flows modeled FDML - Low level workflow engine - Support user interactions

4.6 Linkages

This section discusses the linkages between the various technology components in the on demand Operating Environment framework as well as the methodology to make effective use of them.

4.6.1 Technology components

To redesign business processes rapidly, several components of the integration framework must work together and integrate in a seamless manner.

Business process choreography involves analyzing the interactions between and within processes in an enterprise to organize them in the most effective manner to achieve the business goals. It requires pulling together various resources of the enterprise (people and applications) in the most effective manner. The appropriate analysis and modelling tools can enhance the productivity of the business analysts performing this analysis and design for the enterprise.

However, these productivity gains may not be fully realized, and time to value for any new product or service is increased, if the results of the analysis cannot be shared with other functions (such as IT) in the organization, and used to deploy the appropriate resources and services. The linkages between the tools are illustrated in Figure 4-42.

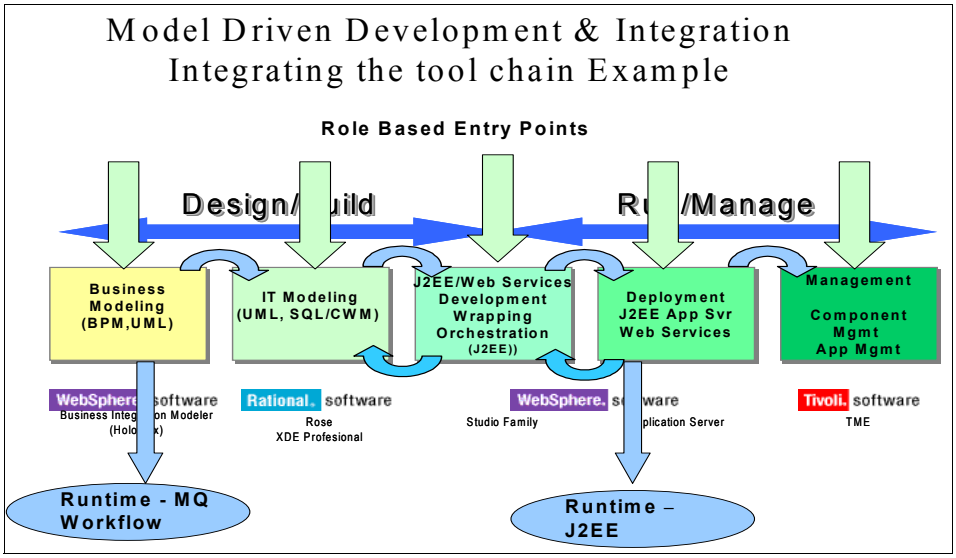


Figure 4-42 Current linkages in the tool-chain

The tools allow the business analyst to model and analyze processes to identify the most effective alternative. Once this has been achieved, the tools can use this information for direct deployment of the model to certain types of run-time, or to share it with other functions in the organization (such as IT). In turn, the tools focused on the IT roles in the organization allow the architecting, building, and deployment of the required “services,” which can be aggregated and exposed to support the business process. As discussed earlier in this chapter, the tools for the various roles have linkages to allow this collaborative flow of information and automation in the generation of services to support the business process. Currently the tools tailored for the IT function, such as the Rational family and WebSphere Application Developer - Integration Edition, are more tightly integrated than those between the business analyst tools and the IT architecture tools (WBI Integrator and the Rational family). These linkages are expected to be enhanced in the future based on open standards such as BPEL and the Eclipse framework. Some of these future capabilities are discussed in “Glimpse of the future” on page 135.

This vision represents a comprehensive end-to-end capability. However, it does not imply only a single entry point, but rather a choice of entry points depending on the organization, skills profile, and infrastructure of the enterprise.

- The business processes designed by the choreography described previously are mapped to actual application and system services by the Enterprise Service Bus, which provides the run-time environment to execute them. It provides common services based on open standards (Web Services) such as security, as well as interfaces for descriptions of the services (WSDL), for their location (UDDI), and for invocation (SOAP). It exposes these services during run time so that the business processes can access and use them with appropriate authorization, and it allows reuse of common components or services within the enterprise.
- Through the standards-based SOA, the Enterprise Service Bus can also be used to rapidly integrate with systems outside the enterprise to provide a seamless view of the extended enterprise to the end user.
- It is impractical to think of a return on investment for a process redesign or rapid time to value, or rapid time to market for new products and services, without leveraging the investment in an organization's existing systems. Adapters provide the linkage for legacy systems to the Enterprise Service Bus. These adapters are based on open J2EE standards such as JCA and JMS.
- A common resource model allows the Enterprise Service Bus to access enterprise information in a consistent manner based on open standards such as Web Services, SQL and so on.

As is evident from the previous discussion, a number of technological and business components must be linked and integrated together to provide the capability for an enterprise to react rapidly and change business processes to adapt to changing market conditions.

4.6.2 Methodology and governance

An overview of this topic was presented in "Methodology" on page 36. This section provides more details on the Rational Unified Process (RUP) tool. RUP is currently used primarily by the IT function in an organization, but is extensible to include the business process analysis function as well. It is expected that RUP will be enhanced to provide greater integration with the business process analysis function provided by WBI Modeler.

IBM Rational Unified Process (RUP)

The RUP platform includes:

- Tools for configuring RUP for a project's specific needs

- Tools for including internal knowledge in process components
- Powerful and customizable Web-based deployment tools
- An online community for exchanging best practices with peers and industry leaders

RUP supports a multitude of technologies, and allows for alternative approaches to be chosen. Any (internal or external) group can extend the RUP process framework with additional guidelines and best practices.

The primary RUP variant uses UML-based modeling for most everything, including business modeling. We haven't done so yet, but it is highly feasible to create a RUP variant that addresses software development processes that use alternative approaches, such as the BPM approach that WBI supports. Many RUP variants already exist that focus on specific tools and technologies. For example, a RUP variant is available that provides guidance on creating systems that are specifically targeted for WebSphere Application Server, using WebSphere Studio as the primary IDE.

RUP is intended to be customized to address a wide range of variations in development tools, run-time technologies, and development processes. Toward this end, RUP is built using a componentized architecture. It ships with tools that can be used to create new RUP components. Other RUP-provided tools can be used to create a RUP configuration that incorporates the new components.

More information can be found on the Rational Developer Network® Web site at:

http://www.ibm.com/software/awdtools/rup/index.html

IBM IGS Method (The Method)

The Method provides a single method to enable a common language among all practitioners delivering business solutions. It's a fundamental component of the shift by Global Services to asset-based services, providing a mechanism for practitioners to reuse knowledge and assets by taking a consistent and integrated approach.

The Method is an integrated method covering:

- Engagement model, which is a theoretical model
- Programming structure and management
- Technical delivery models and techniques

The Method is a work-product-based technical method for IBM Global Services practitioners. Work products, which form the building blocks of The Method, can be arranged, assembled, and custom crafted to support the implementation of industry solutions.

Work products are tangible artifacts that are produced during the project. They include models, reports, diagrams, plans, code, and other documents that are direct stepping stones to the final deliverable. They have a specific purpose in the engagement and describe specific content using predefined semantics and syntax. Work products are produced as a result of performing one or more tasks.

The Method includes both project management and technical methods. The technical methods consist of plan templates and techniques to support a wide range of projects.

4.7 Glimpse of the future

We have highlighted in this chapter two important elements of the integration component of the on demand Operating Environment: the integration between business modeling and IT tools, and the Enterprise Service Bus. We expect many advances in these areas in the near future.

4.7.1 Integration between business and IT tools

As IBM moves to provide a seamless end-to-end tooling solution, the connection between business-level and IT-level tooling becomes vital. Each area has its own integration challenges: the business tools have to address the ease-of-use and non-developer focus that has been a hallmark of the WBI Modeler product to date as they move to an Eclipse base; IT tools have to evolve as we further segregate them into direct-to-middleware tools with very domain-specific languages and those based on the more generic UML.

The ultimate objective is to provide tooling to the business user that focuses on the business aspects: strategy, process, organization, and information. Likewise, IT tooling will focus on technical and platform aspects such as choreography, security, and data management/integration. In this regard, the convergence of our tooling implementation around the Eclipse platform and common meta-models will allow for these tools to exchange information and models both at and between the levels. We expect that the ability to generate BPEL choreographies from a business process model will be an intuitive task and allow enterprises to leverage the advanced capabilities present in the IBM run-time platform.

In the standards arena we see a number of key activities. We have mentioned BPEL already, and with that comes the WS-* family of Web Services specifications for security, integrity, policy, and so on. More importantly, in the modeling area we see the OMG finalizing version 2.0 of the UML, as well as specifying a subset of the UML as the Business Process Meta-model, allowing tools to reuse the semantics of the UML in the development of business-level

tools. In this area, IBM is leading the development of the standards based upon our experience in doing exactly this in the development of future versions of WBI Modeler. We also see the OMG working with BPMI.org in the future development of the latter's Business Process Modeling Notation (BPMN). This is expected to be presented as an alternate notation for the same subset of the UML being submitted under the BPD banner.

It is IBM's intent to use such standards wherever possible. However, in the development of business-level tools, especially those targeted directly to the business user, it is important to focus on ease of use, simplicity, and expressive power. In this regard, the specific notation for process definition, for example, will probably be a subset of the capabilities of either the UML or BPMN notation to allow customers to refine such models in detailed tooling in the IT-level.

4.7.2 Enterprise Service Bus

Standards-based messaging is a critical foundation for a successful ESB. As the world moves toward a Web Services orientation, the ability to leverage Java messaging service (JMS)-based message-oriented middleware (MOM) will be important as companies look for ways to move information quickly and efficiently through a standards-based, Internet-oriented computing structure. This type of middleware provides store-and-forward, event-driven, and guaranteed delivery of data crucial to the success of distributed IT.

Application servers like WebSphere will soon provide a complete infrastructure for the Enterprise Service Bus that leverages Java messaging services and introduce additional protocol support, dynamic selection of the best available service endpoints, management of defined policies and service level agreements, as well as shared mapping, transformation, logging, and other auxiliary services.

4.8 Summary

This chapter described an enterprise environment that can react rapidly to a changing business environment. It discussed the elements of the on demand Operating Environment integration framework that apply to this class of business problems, and how it allows a business to achieve rapid time to value.

The Operating Environment allows business processes that include people and applications to be modelled, simulated, and rapidly implemented on the IT infrastructure. It allows the system to be measured effectively in terms of business metrics and value. It is set up to allow an evolutionary path while preserving investment in existing infrastructure, skills, and applications.

The elements of the Operating Environment were then discussed in the context of two practical business scenarios.

The tools and infrastructure discussed in this chapter provide significant value today. The longer term vision is to enhance them to provide greater integration between the various elements.

The on demand Operating Environment framework addresses the technical elements required to have a successful project. Given the number of organizational functions involved in such an effort, the availability of integrated technology tools and infrastructure represents only a piece, albeit an important one, to enhance the probability of success. An appropriate governance model and methodology are equally important and often overlooked elements of the picture, especially as project and organizational complexity increases.

5

How to react in real time through seamless flow of information

This chapter discusses an approach by which customers can leverage an on demand Operating Environment to react in real time by ensuring a seamless flow of information in the extended enterprise.

After introducing and positioning the technologies that provide this capability, this chapter describes a scenario to highlight the concepts as they may apply to enterprises. It also discusses the linkages between the products and some future trends and directions.

5.1 Introduction

An on demand business must be able to find and capitalize on the value of information, across its business and independent of how it is stored. Businesses must enable information to flow through the business processes within the enterprise. Information, in all forms, is critical to the continuation and completion of business processes to achieve objectives. Further, businesses must make this information accessible to employees, customers, partners, and suppliers in a way that makes it simple to interpret and take action on in real-time.

Enterprises have significant investment and value in the “legacy information” in an enterprise. Users require the flexibility to define what kind of information they are interested in, independent of where the “real” information is stored or what application generates and maintains it.

To ensure seamless flow of information it should be easy to interpret. Fields such as date, time, name, and more can all be compared and viewed in the same way. With normalized data fields, analytics can be applied to that information to assist with more complex interpretation, like identifying patterns in the data.

Briefly, the components of the framework that apply to information integration are:

- **Access and collaboration:** As information flows seamlessly through the business process, many individuals need to interact with that information in different ways. The information must be presented to the user in a personalized manner depending upon their role and the task they are performing. This presentation must be dynamic, allowing changes to be presented in real time. In addition, the tools used must mirror the dynamic and ad hoc nature of collaborative and project teams as they come together to work on a particular problem or issue that may represent a step or task in a larger work flow or decision-making context. These collaborative and ad hoc workflow capabilities are key to providing flexibility to the people in an on demand business as they complete the tasks in a business process.
- **Business process execution:** This includes the tools and linkages to allow various users, such as business analysts, data architects and administrators, and IT developers to model the information required to make decisions, and design, build, and deploy systems to provide this information more effectively. It is a role-based approach with effective flow of information from one role to the other, as appropriate for the specific business issue being addressed. This must be provided in a flexible and rapid manner, with enterprise-level information available to enhance decision making.
- **Enterprise Service Bus:** This provides a mechanism to connect the various applications and information sources in an enterprise in a manner that can be

mapped to the business process being solved. It is based on open standards and allows effective connectivity of applications both within an enterprise and outside it to partner systems.

- **Common information and resource model:** Provides the foundation for a consistent way of viewing and accessing information across the enterprise and with its business partners.

5.2 General strategy

The general plan for this approach is as follows:

- Analyze and model the information requirements for the extended enterprise.
- Determine the information integration approach, such as federation, caching, and so on.
- Develop business metrics to measure effectiveness of the system.
- Architect the federated system (if appropriate) and develop it based on open standards so that it can integrate with the Enterprise Service Bus. The required information is provided and accessed as a “service.”
- Plan and develop the user interface, personalized based on role, with collaborative and ad hoc workflow capabilities.
- Test, deploy, and monitor the system.

5.3 Solution components

This section provides more detailed information on solution components.

5.3.1 Access and collaboration via portal technology

Portals provide the user with a single point of access to a wide variety of content, data, and services throughout an enterprise. The content displayed in portlets on the portal page can be personalized based on user preferences, site design, and marketing campaigns.

A portal delivers integrated content and applications, plus a unified, collaborative workplace. Indeed, portals are the next-generation desktop, delivering business applications over the Web to all kinds of client devices.

How do portals enable on demand?

Portals provide users access to information on demand. Instead of a rip and replace strategy, portals provide the ability to link together disparate, distributed,

heterogeneous systems. Portals can ease the integration burden and tie new products and technologies into existing infrastructures easily and at a low cost.

Portals provide the ability to integrate and communicate with key suppliers and distributors outside the company.

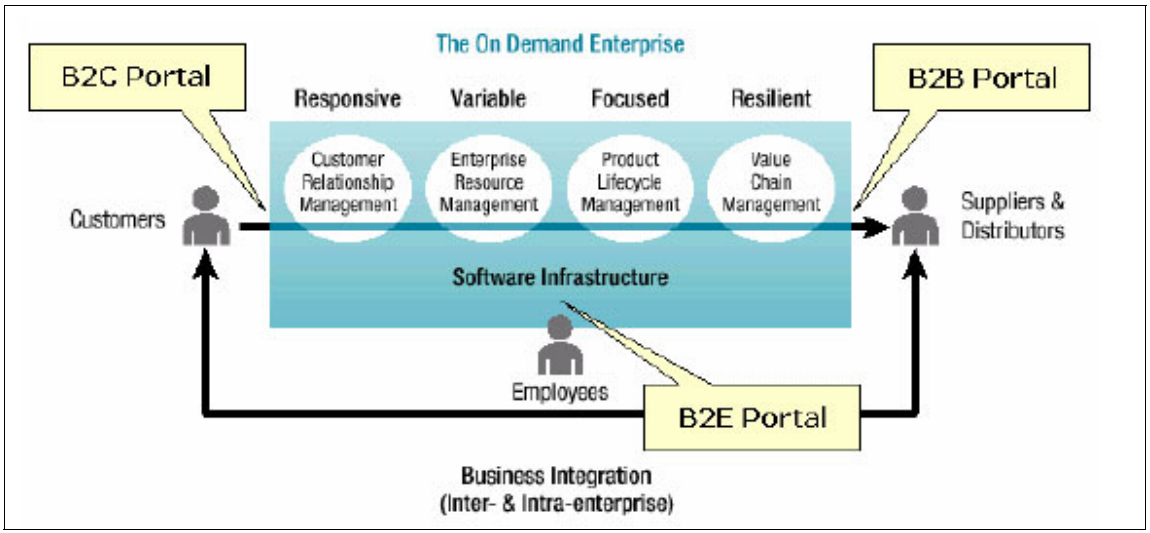


Figure 5-1 Portals in an on demand environment

5.3.2 Access and collaboration via Lotus Workplace

The *workplace* role in an on demand enterprise is crucial: it is about integrating and collaborating. Processes and applications don't make decisions, people do. Responding with speed and building an on demand business requires the collaboration of human interaction with processes and information. These are the capabilities that let people get the just-in-time advice, education, consensus, and approval they need to respond quickly, in the best possible way, to any business situation or emergency.

Lotus Workplace takes all of the Lotus collaborative products—years of investment in messaging, e-learning, calendaring and scheduling, awareness, e-meetings, team rooms, workflows—and makes them accessible through a single, role-based portal where they can be integrated with each other and with other applications, and used in context with the task at hand.

While the workplace framework can be customized to any business needs, there are workplaces focused on capabilities or particular industries such as financial services, insurance, or manufacturing.

How do workplaces enable on demand?

By integrating people, information, and business processes, Lotus Workplace enables people to be more responsive and efficient when handling customer, business partner, and supplier inquiries. People are also able to make informed, fact-based, timely decisions, always collaborating and interacting in line with business processes and best practices. What's more, people are able to build resilient relationships and linkages with other people or organizations, while developing skills needed to transform their business, advance their strategy, and give them competitive advantages.

Workplace components

Workplace components provide a variety of functions, among which are:

- e-mail
- Group calendaring
- Instant messaging
- Conferencing
- Team document sharing
- e-learning classrooms

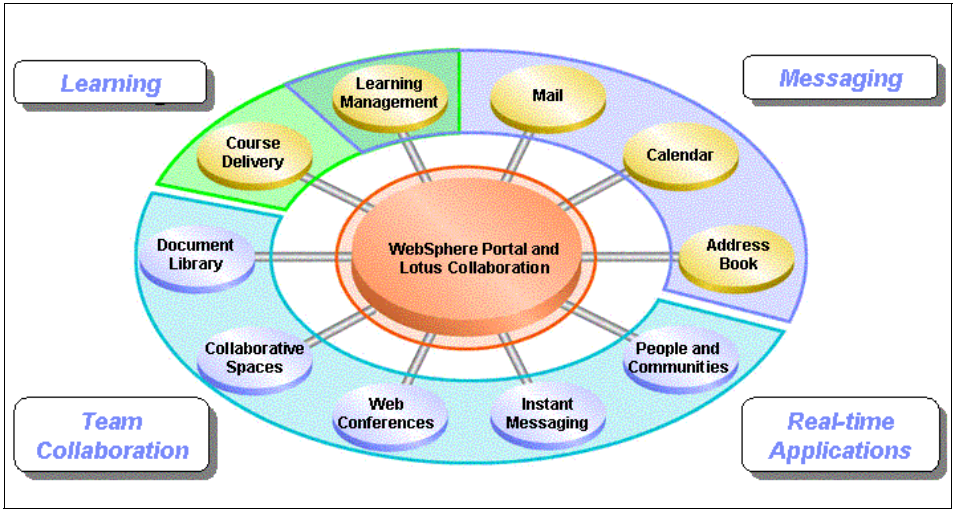


Figure 5-2 Lotus Workplace components

Lotus Workplace packaging

The 1.1 version of Lotus Workplace includes four products: two existing ones and two new ones. They are briefly described here.

- **IBM Lotus Workplace Team Collaboration 1.1** combines instant messaging and awareness, Web conferencing, team workspaces and threaded discussion capabilities to help dispersed teams drive projects to completion.

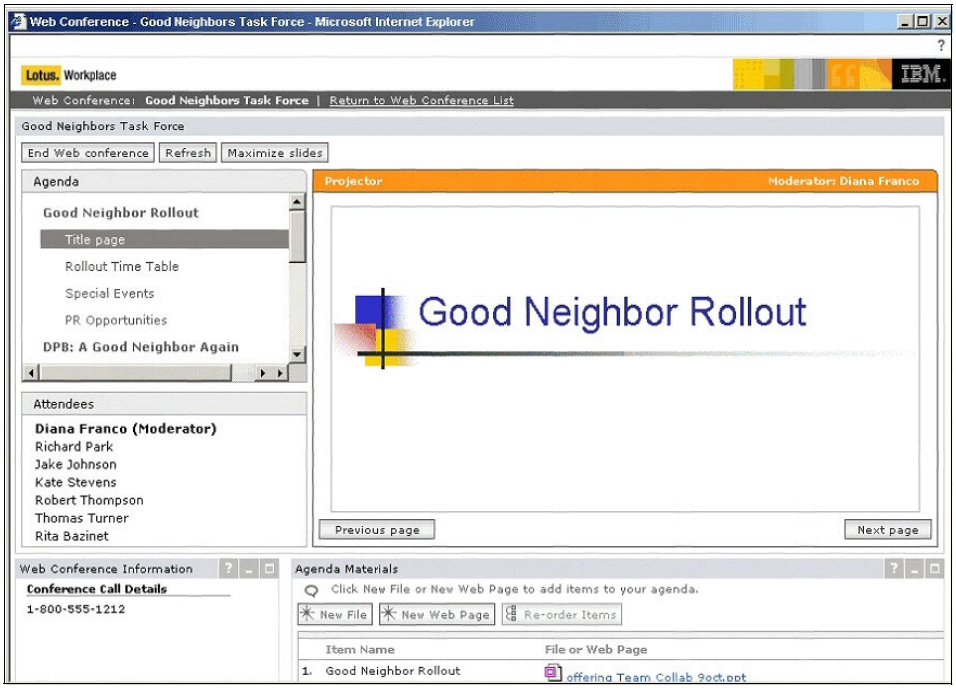


Figure 5-3 Lotus Workplace Team Collaboration

- **IBM Lotus Workplace Collaborative Learning 1.1**, the newest release of the award-winning IBM Lotus Learning Management System, is enhanced with improved functionality and support for the Lotus Workplace environment. IBM Lotus Workplace Collaborative Learning offers a portlet-based user interface (UI) that seamlessly integrates online learning resources on the desktop.

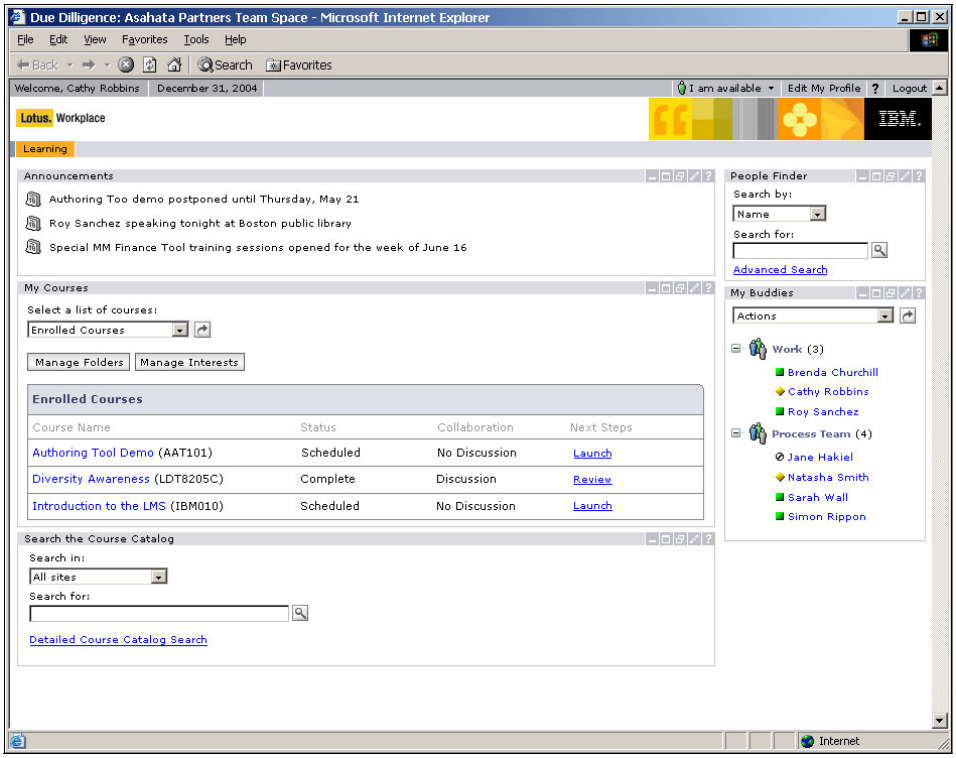


Figure 5-4 Lotus Workplace Collaborating Learning

- **IBM Lotus Workplace Messaging™ 1.1** is a standards-based, secure, scalable and easy-to-deploy solution that lets a company extend it's existing messaging infrastructure to previously "unserved" users, for a remarkably low cost.

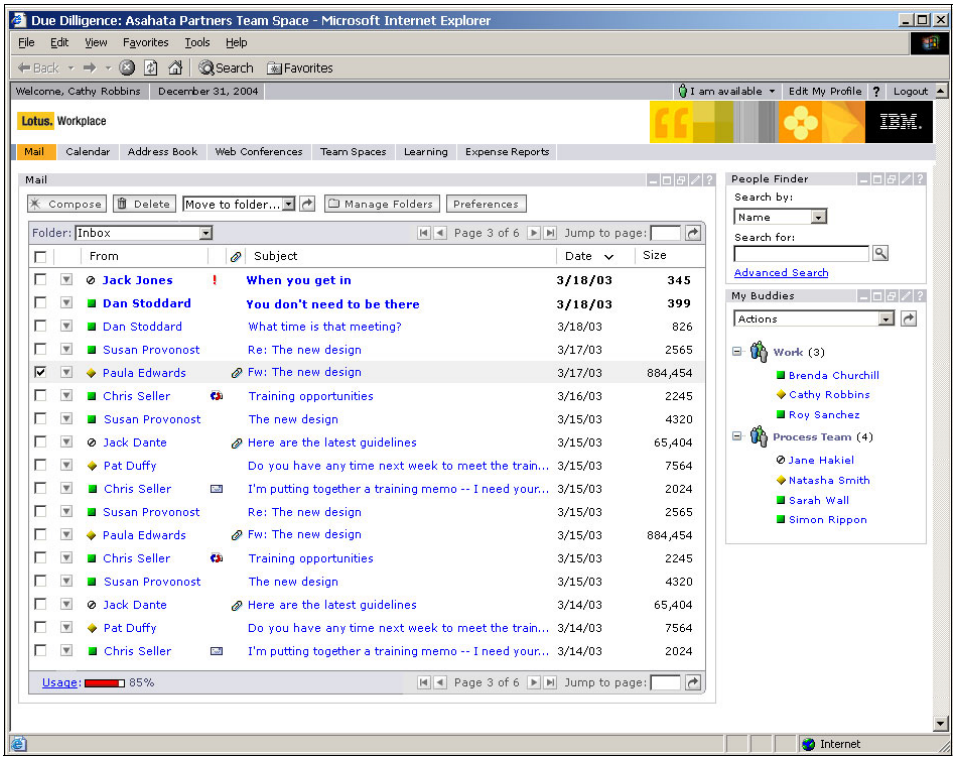


Figure 5-5 Lotus Workplace Messaging

- **IBM Lotus Workplace Web Content Management 1.1** provides templates and automated services that content creators without technical skills can use to access and collaborate on content, build workflows for approving content, lay out pages and links between pages, and properly store and archive content once it expires.

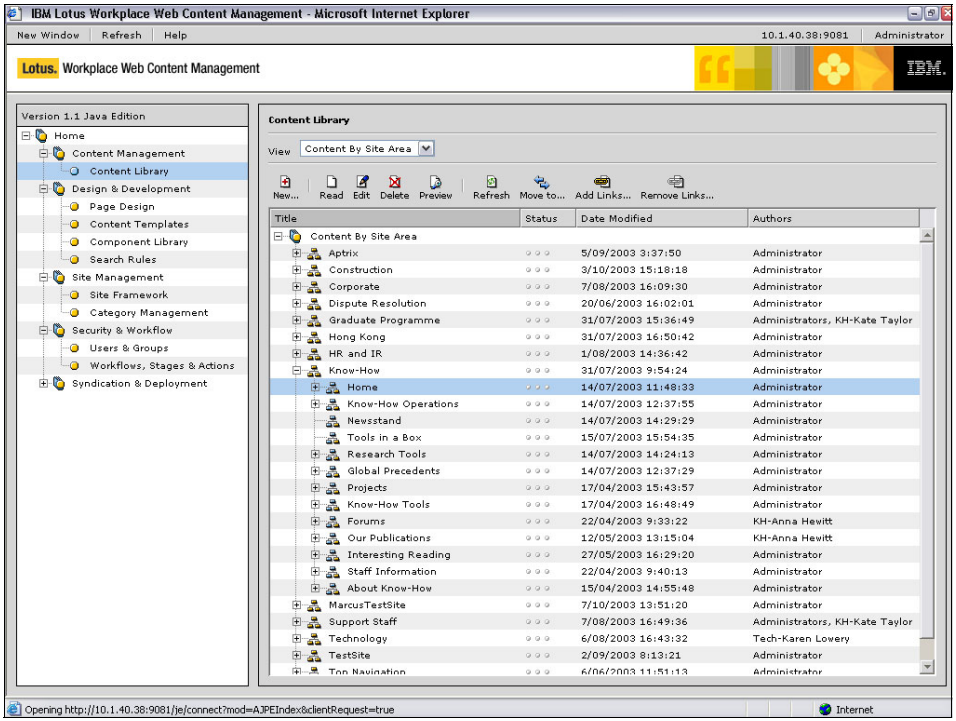


Figure 5-6 Lotus Workplace Content Management

Comparison of Lotus Workplace and WebSphere Portal roles

Both the Lotus Workplace and WebSphere Portal products fit within the integration component of the on demand Operating Environment. The WebSphere Portal is intended to provide integration of people, processes, and information by consolidating access to the necessary applications and data through a single portal interface. The Lotus Workplace family of products focuses more specifically on the collaboration requirements between individuals. It allows for interactions through messaging, shared content management responsibilities, calendaring, and so on.

Within an information flow context, they both have their places. For instance, the WebSphere Portal can help a user access and view information from a variety of sources while the Lotus Workplace products allow for multiple users to access, share, and collaborate on available information.

5.3.3 Business process execution

In the context of information integration, *business execution* defines how the information required for business functions in an extended organization is created and organized to present a consistent and unified view. This allows the processes and people that require information to use it in near real time to enhance collaboration and decision making, which in turn enhances the business flexibility of the organization.

This choreography represents the collaborative effort of various functions in an organization to rapidly model the information needs, and deploy them to a run-time IT infrastructure efficiently. As in process-based choreography, this involves various functions in an organization.

These roles are shown in Figure 5-7.

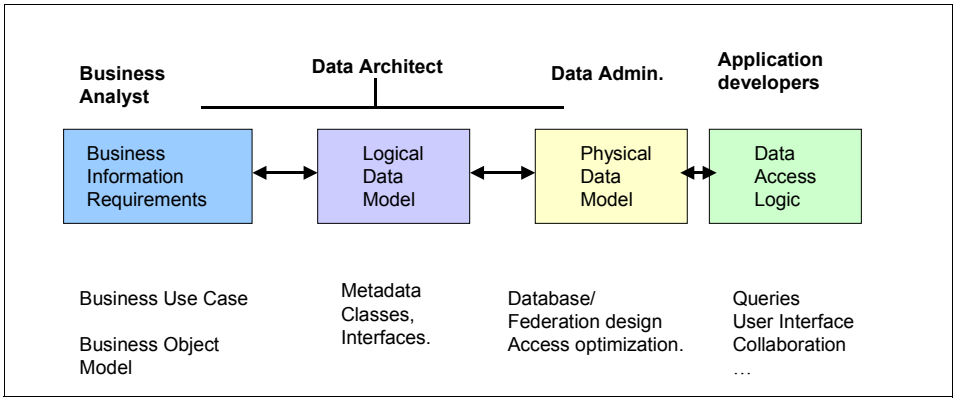


Figure 5-7 Roles in information integration

The process of defining the information requirements for a business objective and designing a unified view of the information in an extended enterprise requires the collaboration of various roles including business analyst, data architect, data administrator, and application developer. Clearly these roles are present in different business functions, such as the line of business and the IT department. Even within the IT department, traditionally the data management function has been distinct from the application development and infrastructure function.

The use of tools can help ensure that the diverse groups can collaborate productively to rapidly provide a unified view of information. Tools also reduce errors by providing an automated flow of information between the various roles, reducing the need for repeated data entry.

Of the various roles outlined in Figure 5-7, the data architect is unique and plays a central role in the development of business requirements, logical information models, and metadata models within the enterprise. The function of the data architect has a well established and rigorous methodology for data modeling to truly reflect business requirements. The use of automated tools for this purpose is also well understood and accepted in the data architect community.

Figure 5-8 shows some of the current tools used by the various roles.

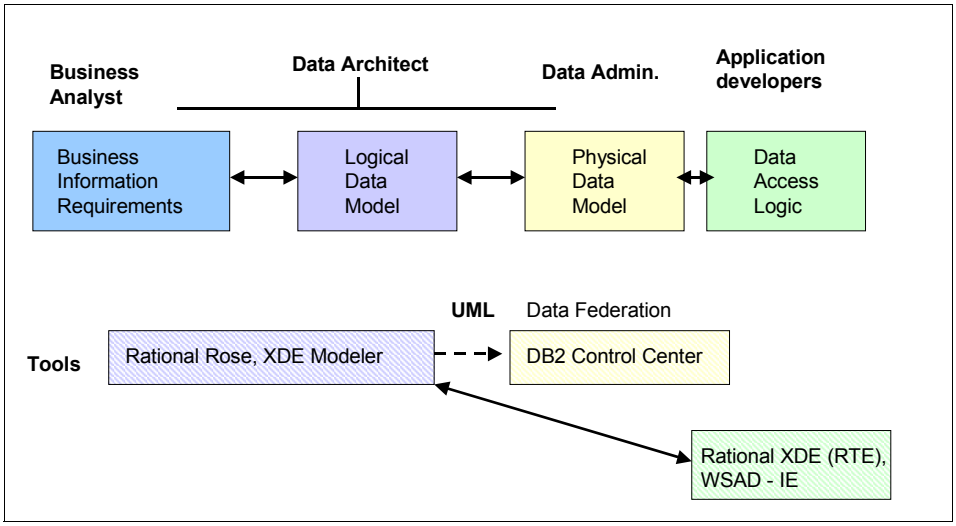


Figure 5-8 Toolsets for roles in information integration

The Rational XDE Modeler and Rose toolsets provide the data architect with a comprehensive modelling capability that includes developing business models, logical data models, and also the physical data models for various databases including DB2.

The linkages between the various stages are provided through UML. The XDE toolset can also be used for reverse engineering, that is, developing the logical data and classes from the physical data. Any changes in one model (physical, logical, or business) made via the XDE toolset are automatically reflected in the others, reducing cycle time and errors.

The XDE toolset also provides a close linkage to the J2EE environment and application developers through its plug-in integration into the Eclipse environment. It provides Round Trip Engineering (RTE) capability with WSAD-IE, where any changes in the application design are automatically reflected in the implementation objects and vice-versa.

While the XDE toolset provides the capability to generate the physical data model for traditional databases, this capability is not currently available for federated data stores. Currently, XDE does not support the “nicknames” for external data stores and therefore cannot be used for setting up the physical model in a federated environment.

For a federated data environment, based on current capabilities, the data architect can use XDE to set up the logical data model, but would need to transfer it manually to the data administration function, potentially as UML diagrams. These would then be implemented by the data administration function through the DB2 Control Center. Similarly, any changes in the federated physical data model and infrastructure would need to be manually inserted into the XDE environment. Refer to 5.7, “Glimpse of the future” on page 204 for a discussion of potential future enhancements to address this capability.

The DB2 Control Center environment provides a graphical, easy to use interface for setting up a federated data environment that provides a single view of the information in an organization and its business partners. It can be used to provide real-time access to this information.

WSAD-IE provides the tool of choice for implementation of the application logic in this environment. It is closely integrated with the DB2 II environment for generation of SQL queries to various types of information sources including relational (SQL), unstructured, XML, and Web Services. Sample screens showing generation of SQL for access to Web Services are shown in 5.4.5, “Solution approach” on page 167.

WSAD-IE also provides a standards-based toolset and environment for development of the portal framework for user interaction and the Lotus Workplace environment to enhance productivity for people-to-people collaboration as discussed earlier.

The on demand Operating Environment provides a toolset enabling comprehensive support for various roles in providing a unified view of the information. This allows businesses to react in real time to the most critical information and provides a seamless flow to enhance collaboration and decision making. The toolset allows for rapid development of applications to aid decision making in support of business objectives, in a manner that reduces time to value, code development efforts, and errors.

5.3.4 Enterprise Service Bus

The Enterprise Service Bus (ESB) provides a mechanism to connect the various applications and information sources in an enterprise in a manner that can be mapped to the business process being modeled. It is based on open standards

and allows effective connectivity of applications both within an enterprise and outside it to partner systems.

For a detailed discussion of ESB please refer the discussion in 2.2.3, “Enterprise Service Bus” on page 17 and 4.3.2, “Enterprise Service Bus” on page 81.

5.3.5 Common resource and information model

Information integration is a category of middleware that lets applications access data as though it were in a single database, whether it is or not. It enables the integration of data and content sources to provide real-time read and write access, to transform data for business analysis and data interchange, and to manage information for performance, currency, and availability.

There are a number of key functions that must be addressed by an information integration platform:

- Read/write access to all forms of information:
  - Structured, semi-structured, and unstructured
  - Heterogeneous: legacy, packaged, and Web
  - Real time versus point-of-time
- Unify query access over local and federated data: SQL/XML and SQL
- Metadata management: Access and manage data and its description with common tools
- Choice of storage modes: Relational or XML
- Choice of data delivery: Application interface as CLI, ODBC, JDBC (relational), Web Services (XML)
- Data placement management: replication, ETL, and caching over heterogeneous data

The IBM DB2 Information Integrator addresses all of these key functions.

Bringing focus to information integration will accelerate development and innovation to deliver solutions that maximize the value of information assets of an enterprise.

The information integration framework’s objective (see Figure 5-9) is to promote simple integration at the data and information layer to be used in conjunction with the application and business process layers of the traditional business integration stack, thereby avoiding the complexities typically associated with having to learn and implement various product APIs in the process of business integration.

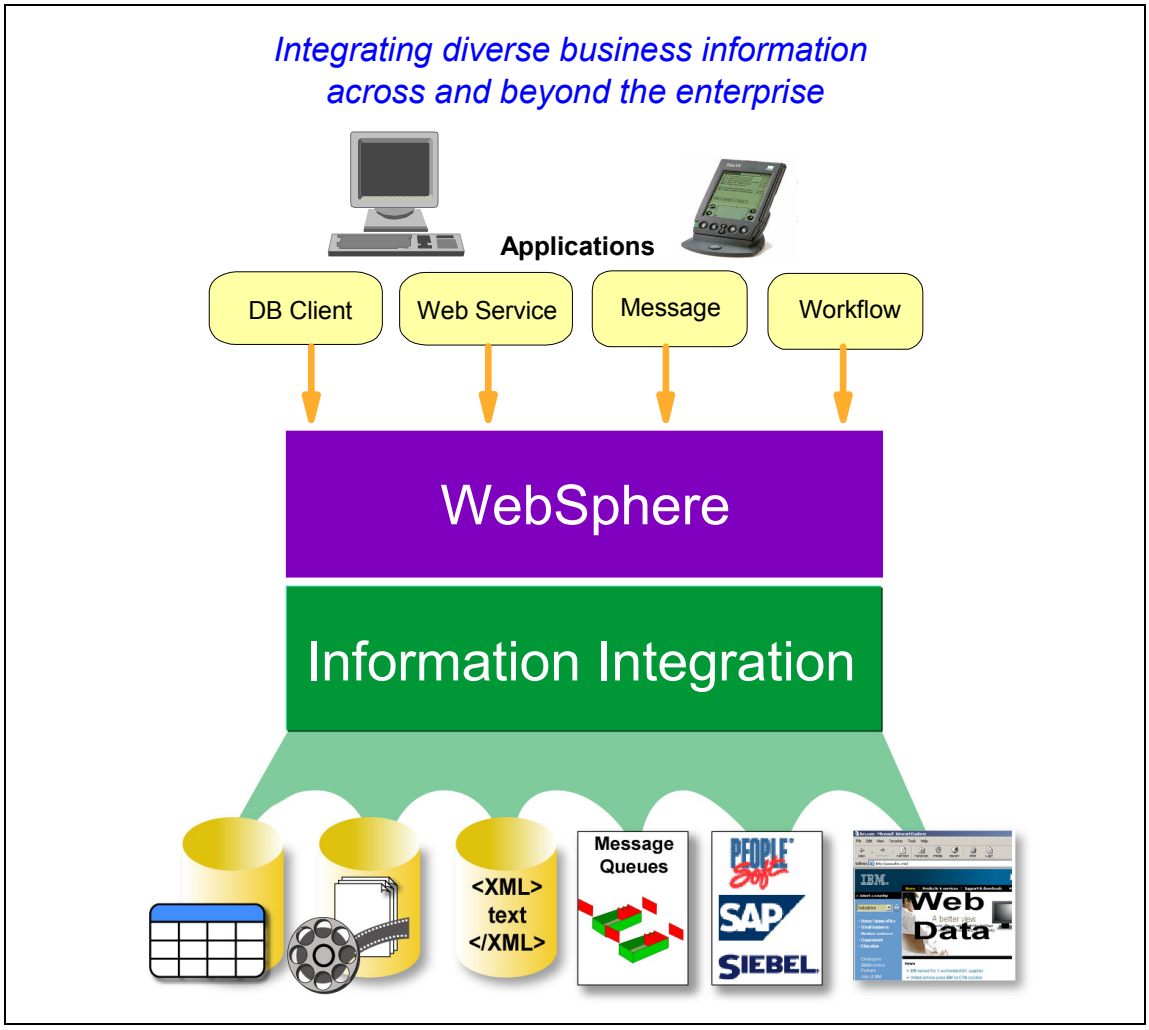


Figure 5-9 Integrating information design stack

The IBM information integration infrastructure will enable integration of diverse, distributed, and real-time data as if it were a single source, no matter where it all resides. The key features of the infrastructure include the ability to federate, search, cache, transform, and replicate disparate data.

Federation

Federation is the concept of gathering a collection of data sources that can be viewed and manipulated as if they were a single source, while retaining their

autonomy and integrity. The resources may be uniform or diverse, collocated or distributed, depending on the implementation.

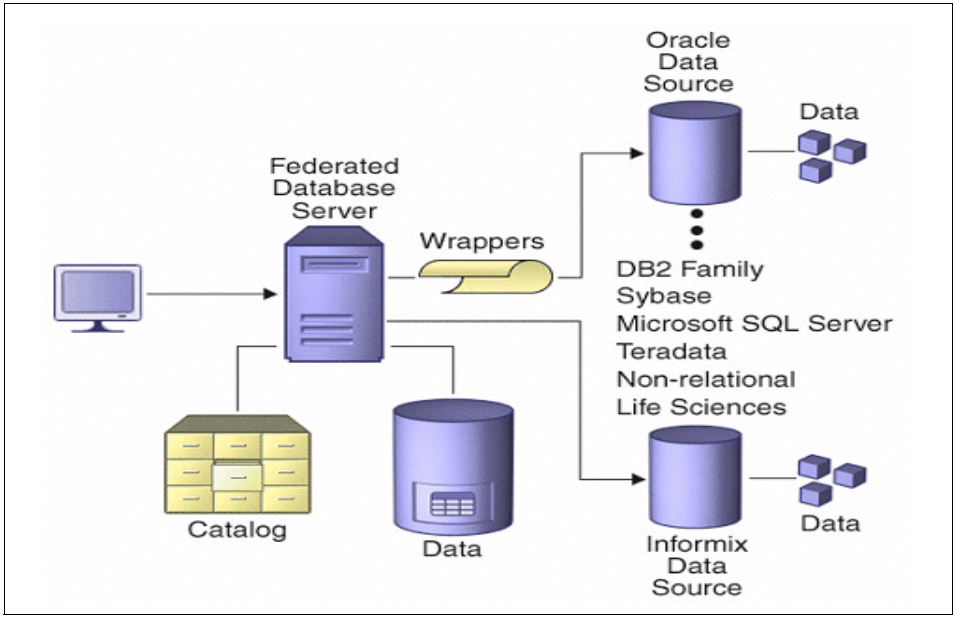


Figure 5-10 Data federation

As seen in Figure 5-10, federation provides the ability to create a single-site image of disparate data by combining information from multiple vendors' databases located elsewhere using database client wrapper libraries.

A "wrapper" is a data federation engine that interacts with heterogeneous sources of data. Wrappers take the SQL that the federation engine uses and maps it into the API of the data source to be accessed. For example, they take DB2 SQL and transform it to the language understood by the data source to be accessed.

The vendor databases may be located locally or remotely and access is optimized using a middleware query processor. A federated server is an abstraction layer that hides complexity and idiosyncrasies associated with the implementation of various vendor databases. The federated RDBMS works behind the scenes to make access to this disparate data transparent and efficient. Such work includes automatic data transformations, API conversions, functional compensation, and optimization of data access operations. Federation also allows the presentation of customer data through a unified user interface.

DB2 Information Integrator, DB2 Information Integrator for Content, and DB2 UDB are the product offerings that provide heterogeneous federation

capabilities. DB2 UDB is the foundation for DB2 Information Integrator and federates data from DB2 UDB on any platform and data from Informix. DB2 Information Integrator federates data from DB2 UDB, Informix, and non-DB2 sources such as Oracle, as well as non-relational sources such as XML. While DB2 Information Integrator and DB2 UDB have an SQL interface, DB2 Information Integrator for Content uses IBM Content Manager interfaces and object APIs as its federation interface.

Getting a unified view of the data through the enterprise can be accomplished using SQL and XML client access to federated databases.

Information integration enables federated access to unstructured data (such as e-mail, scanned images, and audio/video files) that is stored in repositories, file systems, and applications. This facilitates efficient storage, versioning, check-in/check-out, and searching of content data. An additional benefit of information integration content management is the reduction in operational costs.

For setting up a federated database, create a new database in a DB2 instance and perform the following steps to gain access to a remote data source:

1. Create a wrapper.
2. Define a server.
3. Define user mapping if necessary.
4. Create and authorize nicknames.

All of these steps can be performed either through the DB2 Control Center or, for example, through SQL in a DB2 command-line processor window.

For details for each step, refer to the IBM Redbook, *IBM Informix: Integration Through Data Federation,* SG24-7032.

The following terms are often used when discussing DB2 Information Integrator.

- **Federated server:** Any DB2 server where DB2 Information Integrator is installed is referred to as a federated server. Existing DB2 instances and new instances created for the federated system can be federated servers.
- **Data sources:** In a federated system there can be different types of remote sources. A data source can be a relational DBMS instance (such as Oracle or Informix) or a non-relational data source (such as Excel). Some data sources can be accessed through other data sources. For example, through the Extended Search data source one can access data sources such as Lotus Notes databases, Microsoft Access, Microsoft Index Server, Web Search engines, and Lightweight Directory Access Protocol (LDAP) directories. Data sources are semi-autonomous. For example, the federated server can send queries to Oracle data sources at the same time that an Oracle application can access these data sources. A DB2 federated system doesn't monopolize

or restrict access to other data sources, beyond normal integrity and locking constraints.

- **Wrappers and wrapper modules:** Wrappers are mechanisms by which the federated server interacts with data sources. The federated server uses routines stored in a library called a wrapper module to implement a wrapper. These routines allow the federated server to perform operations such as connecting to a data source and retrieving data. Usually only one wrapper is created for each type of data source. For example, to access three DB2 for z/OS database tables, one for DB2 for iSeries table, two Informix tables, and one Informix view, only requires the creation of two wrappers: one for the DB2 data source objects and one for the Informix data source objects. Once these wrappers are registered in the federated database, they can be used to access other objects from those data sources.
- **Server definition and server options:** After wrappers are created for the different data sources, the federated instance owner defines the data sources to the federated database. For relational data sources only the connection-specific information is required. For example, a connection to a particular Informix source is specified through the name of the remote Informix server and the name of the remote database. Some of the information within a server definition is stored as server options. Server options are generally set to persist over successive connections to the data source, but can be set or overridden for the duration of a single connection.
- **User mapping and security:** In a federated environment the users or applications are only connected to and authenticated at the federated server. When the federated server needs to push a request to a remote data source, the server must first establish a connection to the data source. For most of the data sources, the federated server does this by using a valid user ID and password to that remote data source. When a user ID and password are required to connect to a data source, an association between the federated server user ID and password and the data source user ID and password must be defined. This association must be created for each user ID that will be using the federated system to send distributed requests. This association is called *user mapping*.
- **Nicknames and data source objects:** After the server definitions and user mappings are created, the federated instance owner creates the nicknames. A *nickname* is an identifier that is used to reference the object located at the data source that needs to be accessed. The objects that nicknames identify are referred to as *data source objects*. Nicknames are different from aliases that exist in DB2. They are pointers by which the federated server references these objects. Nicknames are typically defined with the CREATE NICKNAME statement. The wrapper provides a default mapping between the data types that are used in the data source and data types that are available in DB2.

When an end user or an application submits a distributed request to the federated server, the request does not need to specify the data sources. Instead, the request references the data source object through the defined nickname. The nicknames are mapped to specific objects at the data source. These mappings eliminate the need to qualify the nicknames by data source names. The location of the data source object is transparent to the end user or the client application.

Suppose that the nickname DEPT to represent an Informix database table called FIN1.PERSON is defined. The statement SELECT * FROM DEPT is allowed from the federated server. However, the statement SELECT * FROM FIN1.PERSON is not allowed from the federated server except in a pass-through session.

- **Pass-through sessions:** One can submit SQL statements directly to data sources by using a special mode called pass-through. SQL statements are submitted in the SQL dialect used by the data source. Use a pass-through session to perform an operation that is not possible with the DB2 SQL/API. For example, use a pass-through session to create a procedure, create an index, or perform queries in the native dialect of the data source. Currently, the data sources that support pass-through support it using SQL. In the future, it may be possible that data sources will support pass-through using a data source language other than SQL.

Replication

Replication is required as a fundamental characteristic of an information integration infrastructure. It complements the distributed access features, enabling management of centralized data stores, and provides the necessary infrastructure for efficiently managing data caches. Data caches (Materialized Query Tables) support placing and managing data at multiple points in the data hierarchy to improve performance and may be defined over the federated data sources.

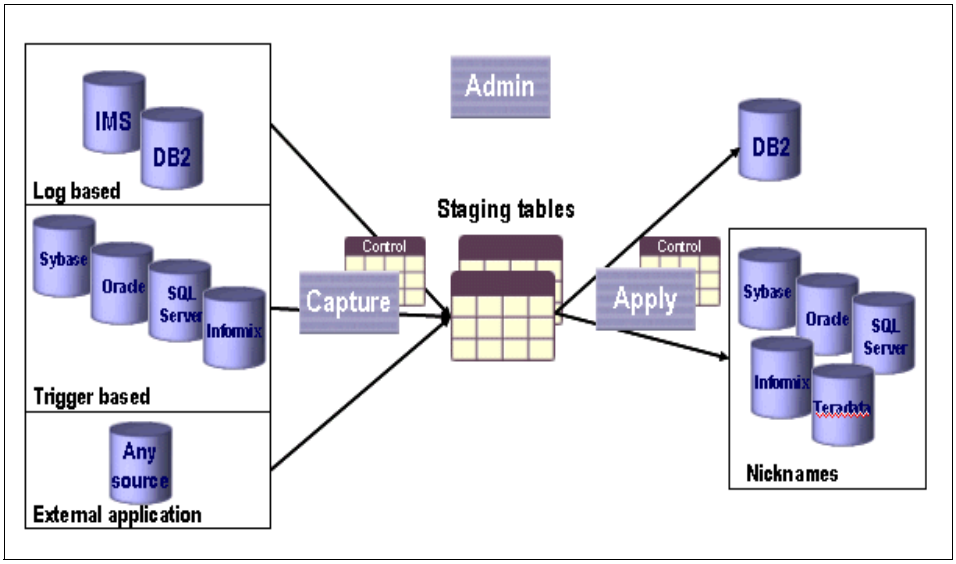


Figure 5-11 Replication architecture

As seen in Figure 5-11, this functionality focuses on efficiently capturing database changes, altering the data to match the target schema, applying the change to the target system, and the end-to-end management of the transfer. This capability allows for either incremental or full replacement in a variety of combinations and with a range of data currency options. It assures that only committed information is captured and replicated. This replication is designed to minimize performance impact to production applications.

Data transformation

The infrastructure must provide rich transformation features to facilitate analysis, interchange, or presentation. Standard SQL includes the ability to apply statistical functions and perform online analytical processing functions. Type-specific features further enhance these transformations, such as applying scoring algorithms, spatial analysis, or chemical similarity searches.

Extensible style sheet language (XSL) translations facilitate document interchange and dynamic style matching to diverse display characteristics. User-defined functions enable customers to standardize any function for any data type. In addition, the ability to access Web Services as a built-in function means that any Web Service can become an embedded transformation function. By leveraging built-in transformation features, data transformation occurs as close to the data store as possible, thereby helping to minimize data movement and speed results.

- Web Services capabilities implemented in information integration include:
  - The ability to expose stored procedure functionality as a Web Service. Using the Web Services Object Runtime Framework (WORF) and Document Access Definition Extension (DADX), these Web Services can be served to clients by an application server (for example, WebSphere Application Server).
  - The ability for stored procedures to become SOAP clients and request Web Services from SOAP servers.
- Advanced search and information mining answer customer needs for sophisticated analysis of the textual information in their applications, content repositories, e-mail repositories, databases, and file systems. Text analytics gather and summarize information about individual documents as well as groups of documents:
  - Language identification describes the language of each document, important for international businesses.
  - Feature extraction identifies information about the document, such as recognizing vocabulary items, names referring to a single entity, locating proper nouns, and abbreviations.
  - Categorization assigns documents into pre-existing categories based on a taxonomy defined ahead of time by the firm (product lines, competitors, and so on).
  - Clustering of information into groups of related documents automatically based on content. This is different from categorization, as it does not require pre-defined classes.
  - Summarization extracts sentences from each document to create a document synopsis.

Data consolidation

Data consolidation (or data placement) physically brings source data together from a variety of locations into one place in advance, so that a user query doesn't need to be distributed. This approach typically uses either extract, transformation, and load (ETL), or replication functionality. As with the federated approach described previously, the end user or application interacts only with this one physically consolidated database server to enjoy the single data source experience.

With data consolidation, both the remote data request and transfer must occur before the end user or application request is issued. It is logical, therefore, that the request to the remote data source is basically formulated one time only during the data requirements definition, while the transfer of data typically occurs many times according to some defined cycle or trigger. Neither the data request

nor the data transfer to and from the remote data source are directly related to the end user's request. Hopefully, there is some relationship with the end user request and the data architect's prediction of the types of queries to be served.

A key consideration for data consolidation is the maximum latency that can be tolerated when transferring the data from source to target. Typically the business needs to specify how up-to-data a copy of the data must be. In data warehouses, for example, the frequency required might be daily or weekly and the latency of data consolidation can easily extend to many hours. At the other extreme, the need for almost real-time data, such as in stock market systems, requires minimum latency in data consolidation.

Two of the important factors determining the latency possible in data consolidation are the complexity of the transformations required and the volumes of data to be transferred. These factors lead to two complementary approaches to consolidating data. ETL is optimized for larger data volumes and often associated with more complex transformations, while data replication emphasizes the transfer of individual data records and is often restricted to simpler transformations.

To federate or not to federate

Data federation and data consolidation are actually similar concepts. Both involve requesting and receiving data that originally resides outside the physical confines of the database server with which the end users or applications interact. The key difference is in the timing of the data requests to, and transfers from, the remote data source and the central server. With data federation, both the remote data request and transfer occur after the end user or application has issued the request to the federated database server.

But from the end users' points-of-view, or that of the applications acting on their behalf, data federation and data consolidation act in opposite ways. Data federation integrates the required information synchronously, directly from its original sources, acting only after the end user decides what information is required. It must therefore return the result in a time frame that is acceptable to the user or requesting application. Data consolidation operates in advance of the user query, allowing itself more time to perform the required processing.

However, the data architect needs to make decisions in advance regarding what data will be required. Data consolidation requires a larger quantity of permanent data storage than the data federation approach because it is creating a second copy of the data.

Preference for data federation approach

Data federation is likely to be the appropriate method of data integration when:

- Real-time or near real-time access to rapidly changing data is required.

  Making copies of rapidly changing data can be costly, and there will always be some latency in the process. Through data federation, the original data is accessed directly and joined in the query. However, the performance, security, availability, and privacy aspects of accessing the original data must be considered.

- Direct immediate write access to the original data is required.

  Working on a data copy is generally not advisable when there is a need to insert or update data, because data integrity issues between the original data and the copy may occur. Even if a two-way data consolidation tool is available, complex two-phase commit locking schemes are required. However, writing directly to the database of another application is generally prohibited.

- It is technically difficult to use copies of the source data.

  When users require access to widely heterogeneous data and content, it may be difficult to bring all the structured and unstructured data together in a single local copy. Or, when the source data has a very specialized structure, or has dependencies on other data sources, it may not be possible to sensibly query a local copy of the data. In such cases, accessing the original source is recommended.

- The cost of copying the data exceeds that of accessing it remotely.

  The performance impacts and network costs associated with querying remote data sets must be compared with the network, storage, and maintenance costs of storing multiple copies of data. In some cases, there will be a clear preference for a federation-based approach, such as when:

  - Data volumes of the original sources are too large to justify copying it.
  - Data is too seldom used to justify copying it.
  - A very small or unpredictable percentage of the data is ever used.
  - Data is accessed from many remote and distributed locations, which would imply multiple copies.

- It is illegal or forbidden to make copies of the source data.

  Creating a local copy of source data that is controlled by another organization or that resides on the Internet may be impractical, due to security, privacy, or licensing restrictions.

- The user's needs are not known in advance.

  Allowing users immediate, ad hoc access to any enterprise data is an obvious argument in favor of data federation. However, care is needed with this

approach. The potential exists for users to create queries, accidentally or otherwise, that negatively impact both source systems and network performance, and that disappoint the user by yielding poor response times.

Preference for data consolidation approach

There are many possible cases where data federation may not be the preferred approach. Data consolidation is more appropriate when:

- Read-only access to reasonably stable data is required.

  The data federation approach will present the remote data in real time. This may not be advantageous to the end user or application, which would prefer to suffer some latency in the data in order to be insulated from the continuous flux of information in the remote operational data sources.

- Users need historical or trending data.

  Historical and trending data is seldom available in operational data sources, but requires a data consolidation approach to build up historical data over time. This is a very common data warehousing requirement.

- Data access performance or availability is an overriding consideration.

  Users routinely want quick data access. Despite the best efforts of a well-designed federated server working in unison with the remote data sources, the volumes of data required may necessitate that a local, pre-processed copy of the data be made available.

- User needs are repeatable and can be predicted in advance.

  When user queries are well-defined, repeated, and require access to only a known subset of the source data, it may be cheaper to create a copy of the data for local access and use. This approach also insulates the remote operational data sources from large, complex, or poorly structured queries.

- Data transformations or joins needed are complex or long-running.

  In cases where significant data transformations are required or where joins are complex or long-running, it is inadvisable to have them run synchronously as part of a user query due to potentially poor performance and high costs. In such cases, creating a copy of the data through data consolidation would seem to be more advantageous.

Using both data federation and data consolidation

It is likely that there will be cases where a combination of data federation and data consolidation techniques is the best option.

One case is where a federated query can leverage data consolidation functionality transparently. This is because sometimes a federated query will not work. It could be because of network outages. Data federation can use data

consolidation to create or manage cached data. On the other hand, data consolidation tools may be optimized for only a subset of available data sources. Using data federation along with it can expand that number, and allow pre-joining of data for a performance impact.

Content federation benefits

The primary benefits of content federation using either DB2 Information Integrator for Content or the DB2 Information Integrator include:

- Transparency

  Content in multiple repositories appears to be in one source. There is no need to move or replicate the data. This is critical due to the high storage requirements of many types of content that should not be duplicated if avoidable.

- Heterogeneity

  Diverse content in the listed data sources. Note that we are focusing on *content* versus *data.* Content in formal content repositories may be better suited to DB2 Information Integrator for Content than for DB2 Information Integrator. Note that both can get to similar content stores since both can exploit IBM Lotus Extended Search, but how the content is used is probably different, and the access path to the content is certainly different. Consider the usage scenario first.

- High function
  - Federated search across content stores.
  - Suitable for dynamic access, such as browsing and discovery. No compilation and code generation for a specific user schema is necessary.

- Autonomy

  Add federation without disrupting existing systems.

- Performance
  - Search is propagated in parallel into data sources.
  - Results come back as soon as any data source returns some hits.

- Flexibility

  Results come back as uniform, self-describing objects, so that they can be used in the next process in the pipeline, for instance workflow or information mining.

The DB2 Information Integrator family of products consists of:

- IBM DB2 Information Integrator V8.2
- IBM DB2 Information Integrator for Content V8.2 (formerly IBM Enterprise Information Portal)

5.4 Scenario

In this section we continue our discussion of the scenario from Chapter 4, “How to rapidly modify business processes” on page 63, regarding a hypothetical retailing enterprise, ABC Electronics.

5.4.1 Business context

ABC Electronics is a retail electronics store that specializes in both consumer and business goods. Founded 30 years ago, the company has grown from a small local storefront to a large regional department store featuring televisions, computer equipment, stereo equipment, and household electronics. The company has a large wholesale business as well, supplying computer equipment, fax machines, copiers, and other business electronics to merchants throughout the region.

As shown in Figure 5-12, ABC Electronics sells to other retailers as well as directly to consumers, through various channels including phone, fax, the Web, and through its storefront.

It has traditionally had close relationships with a set of suppliers, such as supplier A. These suppliers have, over time, treated ABC Electronics as a preferred customer, assuring them of product availability during peak demand times and also extending favorable credit options.

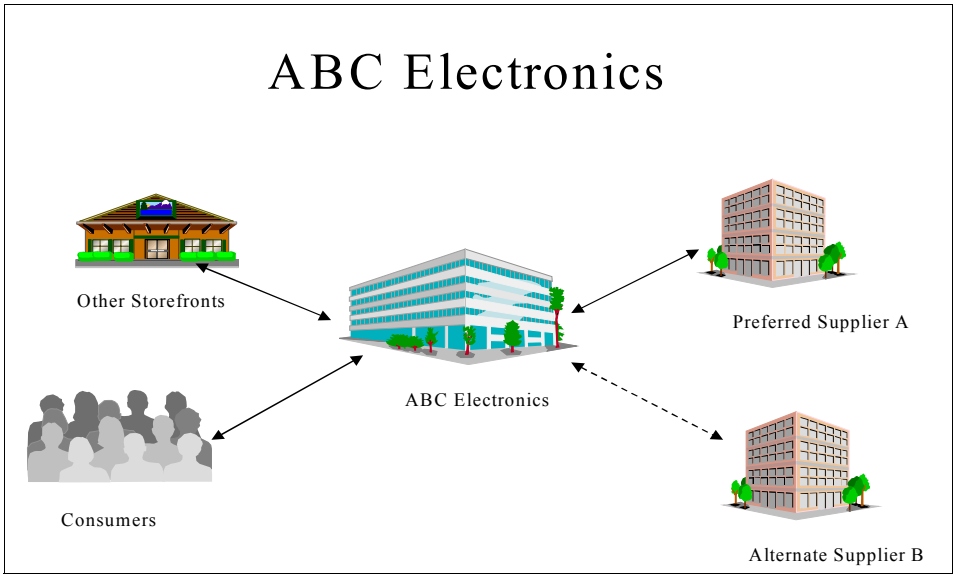


Figure 5-12 Business context

ABC has recently integrated its retail and wholesale operations for greater efficiency and to reduce loss of sales due to out of stock situations.

The company would like to further enhance their inventory management practices by providing internal analysts and their preferred reseller partners a real-time view into the stock levels for various items, at ABC as well as through the supplier partner network. In this manner, ABC can leverage its customer and partner relations to further enhance sales and can reduce inventory costs.

5.4.2 Current environment

ABC has recently been through a redesign of their inventory replenishment process as described in 4.4.2, “ABC Electronics scenario” on page 116. This involved a collaborative effort between the lines of business—retail and wholesale—and the IT organization. The current applications are shown in Figure 5-13.

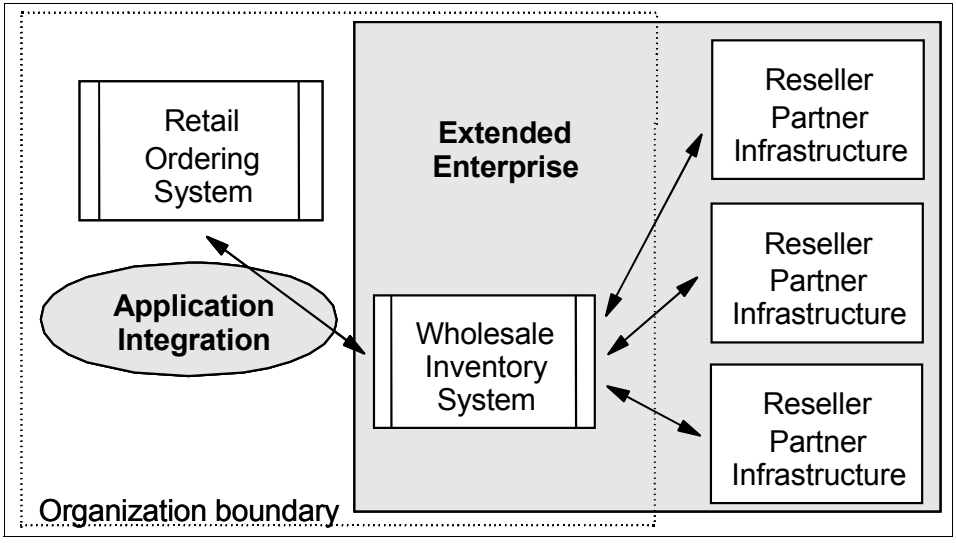


Figure 5-13 Current applications

In order to provide the integration between their retail and wholesale systems, as well as with their reseller partner systems, ABC set up a services-oriented architecture as described in 4.4.2, “ABC Electronics scenario” on page 116. They leveraged their investment in WebSphere as shown in Figure 5-14.

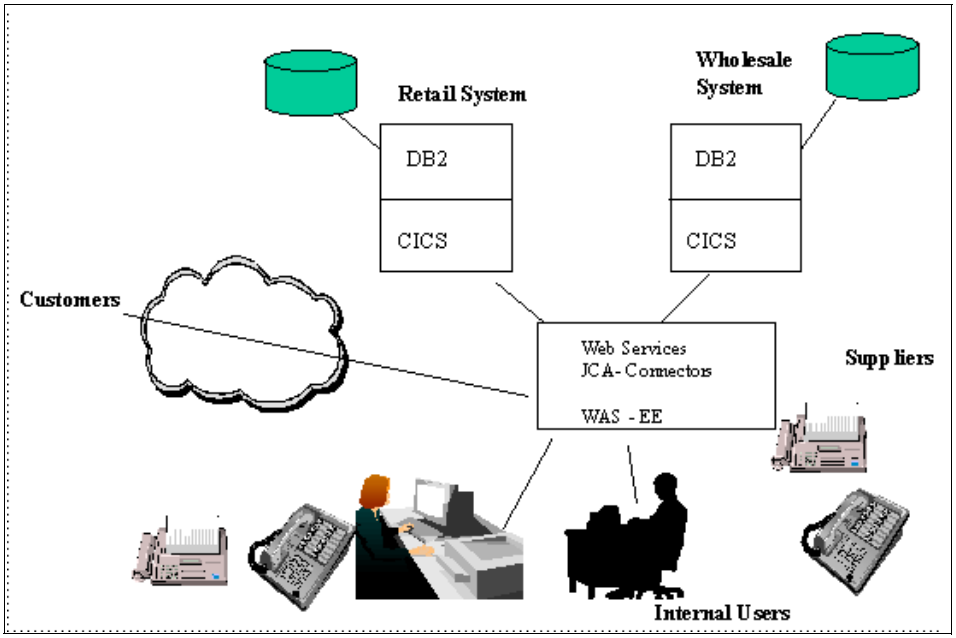


Figure 5-14 Current functional layout

5.4.3 Business objectives

Building on their success integrating internal retail and wholesale systems, ABC Electronics would now like to further enhance their inventory management capabilities by providing a real-time view into the inventory of required items across their internal warehouses, as well as those of their supplier partners.

The business drivers for these objectives are:

- Cost reduction due to more efficient inventory management decisions: prices, dates, stock on hand
- Additional revenue through sales obtained by linking reseller partners with supplier partner information

Other business drivers are the same as in “Business objectives” on page 119.

5.4.4 Technical objectives

In order to achieve their business objectives, ABC would require an IT infrastructure that provides the most effective manner of integrating information inside the enterprise as well as with partners. This information can then be

displayed in a customized manner based on the profile of the end user and their requirements.

Functional objectives

As a first step, ABC would like to get a unified view (query only) for the inventory on hand for a given item, as well as the inventory available from their supplier partners. This information will be used by internal inventory analysts and ABC's reseller partners.

As in the previous discussion, by basing the solution on open standards, ABC will have the flexibility to enhance system functionality in the future, such as adding the capability for reseller partners to directly order from supplier partners via the Web with no intervention from ABC.

The first step (query only) is illustrated in Figure 5-15.

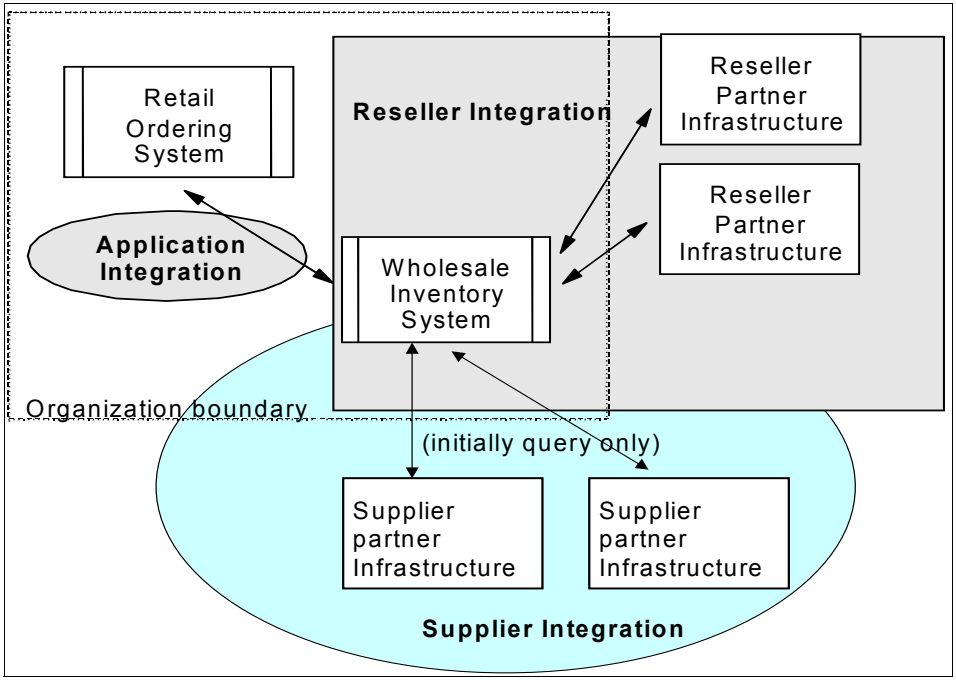


Figure 5-15 Functional requirements

The functional requirements are shown as use cases in Figure 5-16 and Figure 5-17.

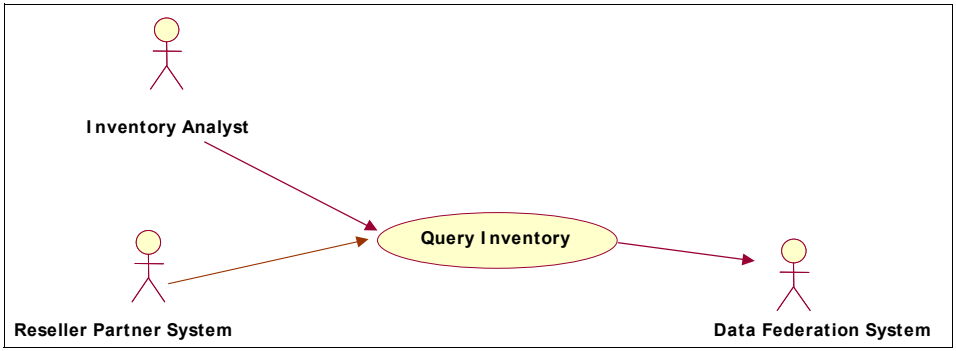


Figure 5-16 Use case for querying inventory

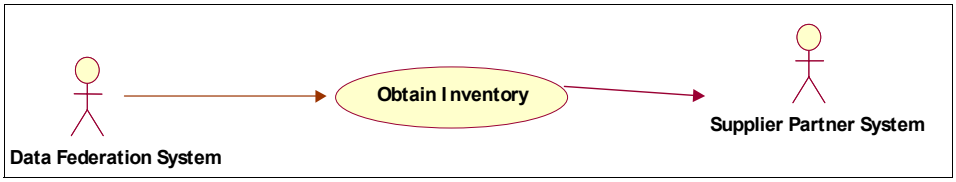


Figure 5-17 Use case for obtaining inventory from partner

Non-functional objectives

From a non-functional perspective, ABC requires that all solutions provide a standard Quality of Service level. The elements of this are discussed in 4.4.2, “ABC Electronics scenario” on page 116. It is beyond the scope of this redbook to define such requirements in real, measurable terms for our sample scenarios. Of course, one would do so in an actual implementation to ensure that the delivered solution meets the demands of the organization.

5.4.5 Solution approach

The first step in developing a solution for information integration is to understand the business information requirements and how they map to information sources.

A team is set up, composed of ABC’s business analyst and representatives from the wholesale organization, as well as members of the IT organization, including the data management function and the application development function. The skills in this team represent:

- Domain expertise in inventory management
- Data architecture
- Data management

- J2EE application development

The team first understands and describes the business requirements, which in this case, map to the business process of obtaining inventory levels for specific items.

- For a specified set of electronic items, the inventory levels, delivery dates, and price are needed:
  - If the request is from a reseller partner process ABC requires:
    - An aggregate inventory level from all of the suppliers.
    - The higher of internal price or the supplier price plus a mark-up, based on quantity requested. Internal inventory should be used first.
    - The delivery date: immediate for internal inventory and longest of supplier dates based on the quantity requested.
  - If the request is from an internal ABC inventory analyst, the same information listed previously is required. However, it should not be aggregated, but rather by line item number by supplier. Further, in the future it can be combined with historical data maintained on each supplier by the wholesale department.
- Supplier partner systems can supply the information through Web Services based integration in XML format.

Based on an analysis of the business processes and the information requirements associated with them, the data architect develops:

- Business use case model, capturing the external business requirements
- Business object model representing internal business requirements.

The data architect uses the Rational Rose and XDE toolsets to represent this in UML.

Functional updates

The function and design are further refined and annotated by the data architect in Rational XDE using UML. The appropriate data sources to be queried are identified, along with their interfaces.

In the case of ABC Electronics, the following databases are required to support the business requirements:

- The Wholesale system. This includes the operational data store as well as a historical data store in which supplier information is stored. This information is stored in DB2 databases. Information about the data model is obtained by the data architect by reverse engineering using Rational XDE.

- Supplier X's system provides a Web Service interface: For a set of item identifiers and distribution partner identifier, it provides the current inventory levels, prices, and delivery dates. Supplier X provides details on the Web Services interface as WSDL. The WSDL will be incorporated into the federated model using WSAD, as described later in this section. While the data architect can develop a UML-based metadata model for information from this source and annotate it, the linkage between Rational XDE and the tools for federated data (DB2 Control Center and WSAD) are currently manual.
- Supplier Y returns the same information as Supplier X, but as an XML file. The Rational XDE tool does not currently provide the capability to reverse engineer the model from an external data source, so the model has to be provided by Supplier Y. The UML-based metadata is communicated manually to the implementation team.

The data architect develops the target logical model, including data classes, types, and so forth. This allows any transformations from the data sources to be determined and also allows the results to be described in XML for output to the requestors.

The implementation team could choose to query each data source individually. However, the time to value, cost, and risk associated with this approach would all be higher because the amount of code developed is higher.

Since in this case the primary requirement is for a unified real-time view of the inventory information, a "federated" data approach could provide more rapid time to value, and lower risk and cost. Further, the skills that ABC has developed in DB2 and the J2EE environment can be leveraged to rapidly deliver an information integration solution based on the DB2 Information Integrator product.

The steps to developing a functional system are identified at a high level in Figure 5-18.

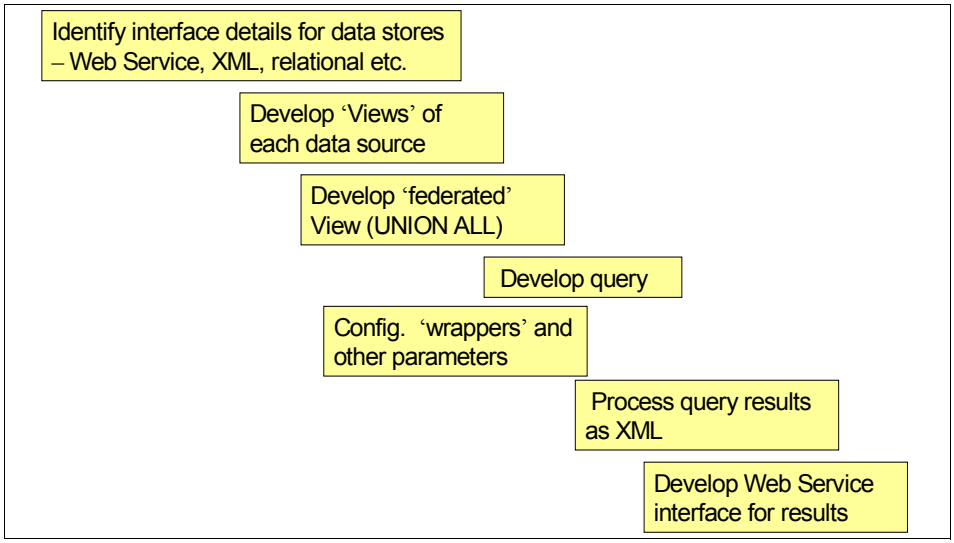


Figure 5-18 Information integration steps

The information obtained from the Wholesale system, Supplier X, and Supplier Y is obtained through a single SQL query based on setting up a federated information aggregation infrastructure, as shown in Figure 5-19.

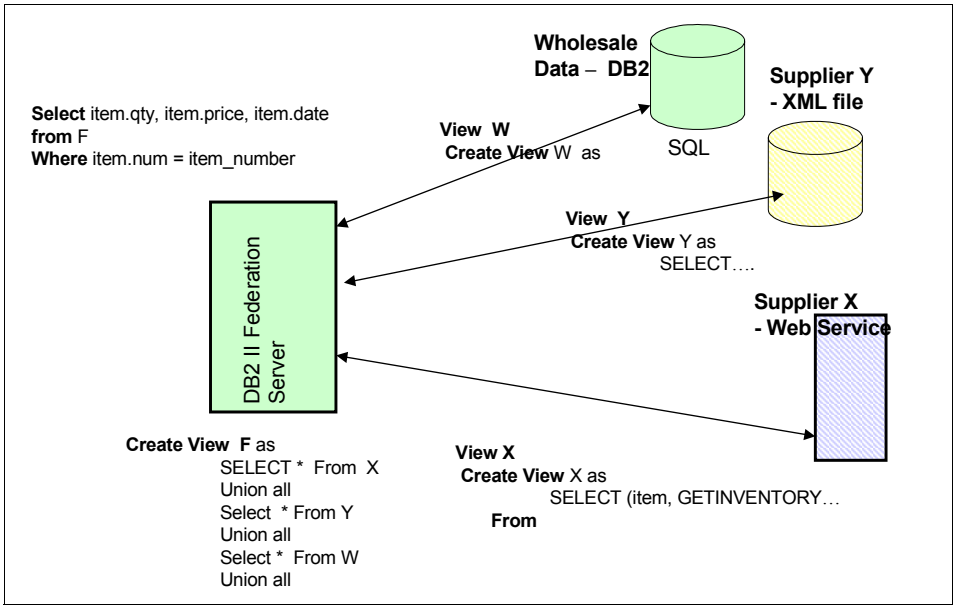


Figure 5-19 Unified view of information

In the case of ABC Electronics, the following databases are required to support the business requirements:

- The Wholesale system database. This is a DB2 database and the data model is available and exposed.
- Supplier X system provides a Web Service. For a set of item identifiers and distribution partner identifier, provides the current inventory levels, prices, and delivery dates.
- Supplier Y returns the same information as Supplier X, but as an XML file.

This information is then wrapped as a Web Service with WORF and provided to the requestor:

- If the request was from a reseller partner, aggregated information is returned in the Web Service as described previously.
- If the information is requested by an internal user, the information user may choose to receive aggregated information or individual line item details.

The high-level information flow is shown in Figure 5-20.

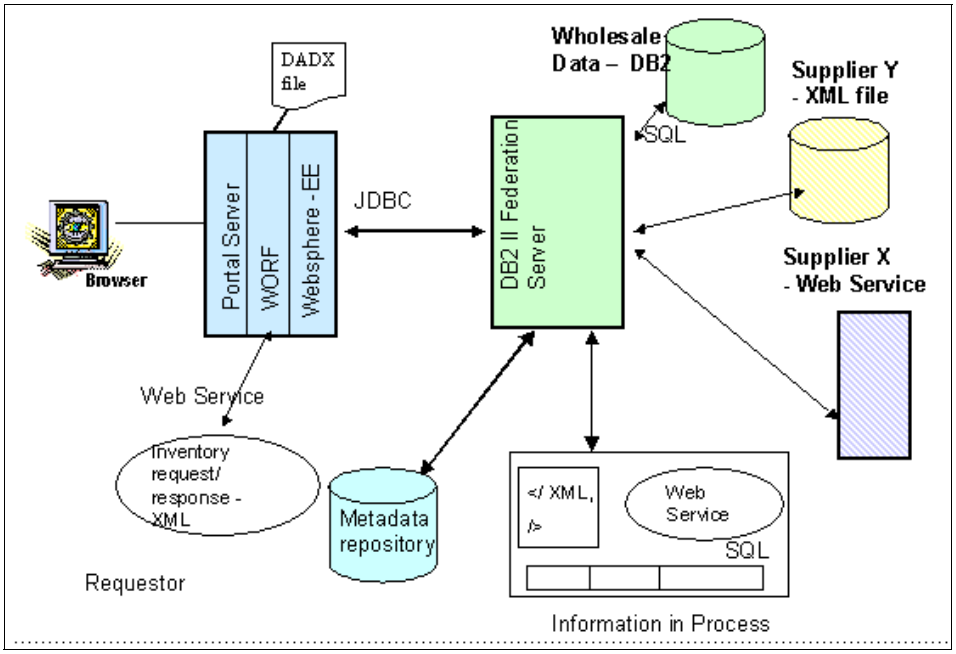


Figure 5-20 Components and information flow

Setting up the federated data infrastructure

Based on the stated requirements, the database administrators jointly refine the logical data model with the business analysts using Rational XDE Data Modeler. Based on the UML for representing the Entity- Relationships in the logical data model, the database administrators can come up with the logical normalized design for the unified database.

As described earlier, DB2 Information Integrator can federate sources that are relational databases, XML documents, or Web Services. The administrative functions and updates to set up this federated infrastructure are handled through the DB2 Control Center, which provides an easy to use graphical interface that leverages ABC's existing DB2 skills.

ABC's data administrators go through the following steps to set up the infrastructure:

1. Create a wrapper. Wrappers take the SQL that the DB2 II federation engine uses and map it into the API of the data source to be accessed. For example, they take DB2 SQL and transform it to the language understood by the data source to be accessed.

   Figure 5-21 shows an example of the DB2 Control Center being used for wrapper administration. The Control Center provides the graphical interface for administration of standard wrappers, and also a plug-in architecture that allows custom wrappers to be administered.

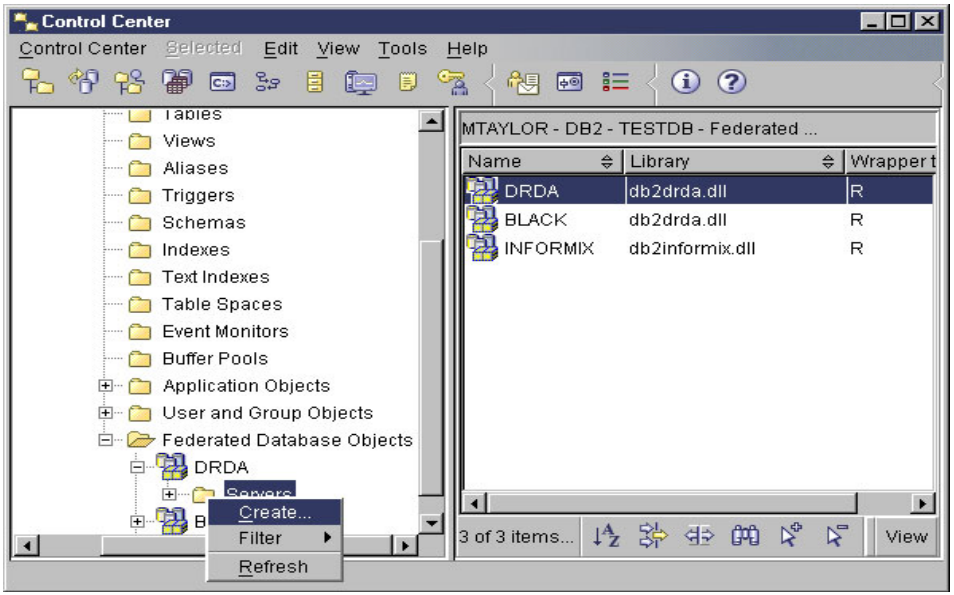


Figure 5-21 Wrapper administration with DB2 Control Center

These wrappers provide four functions:

- Data modeling
  - Map data model to relational data model (tables with rows and columns)
  - Map functions into SQL operations
- Query planning
  - Represent data source capabilities
  - Push down as much work to data source as sensible
  - Detect missing functions at source (so engine can compensate)
  - Supply cost and cardinality information
- Connection and transaction management
- Query execution and data retrieval
  - Execute parts of a user's query for a specific data source

2. Define a server for each data source. After wrappers are created for the different data sources, the data sources are defined to the federated database.
3. Define user mapping for authorization. In a federated environment the users or applications are only connected to and authenticated at the federated server. When the federated server needs to push down a request to a remote data source, the server must first establish a connection to the data source. For most of the data sources, the federated server does this by using a valid user ID and password to that remote data source. When a user ID and password are required to connect to a data source, an association must be defined between the federated server user ID and password and the data source user ID and password.
4. Create and authorize a nickname for the remote database. A nickname is an identifier that is used to reference the object located at the data sources that will be accessed. The objects that nicknames identify are referred to as data source objects. Nicknames are different from aliases that exist in DB2. They are pointers by which the federated server references these objects.
5. Install and configure the components of WebSphere Application Server EE (packaged with DB2) that allow Web Services to be constructed from the XML results of the federated query. These are the WORF and DADX components in DB2 II. ABC's skills in WSAD can be leveraged to set up WORF and DADX components.

For additional details refer to the IBM Redbook *Getting Started on Integrating Your Information,* SG24-6892.

Creating queries and views

Once the federated infrastructure has been set up as described, the integrated information can be accessed by a single SQL query.

There are several tools available for creating the queries that can be used by the DB2 II Federation Server.

ABC Electronics has invested in J2EE and WebSphere skills. With the close integration between DB2 II and the WebSphere Application Server, that can be leveraged in this situation through the WebSphere Studio Application Developer - Integration Edition. WSAD provides a graphical point-and-click user interface to develop functions that can call Web Services queries and generate the code associated with them, based on a wizard.

The ABC data analyst will need to supply the Web Service Description based on a WSDL file or URL, and other information such as the mapping of WSDL elements to the schema, into the wizard. The result is a scalar or table function that returns data directly to the application. The appropriate SQL calls are automatically generated.

Two representative views of this are shown in Figure 5-22 and Figure 5-23.

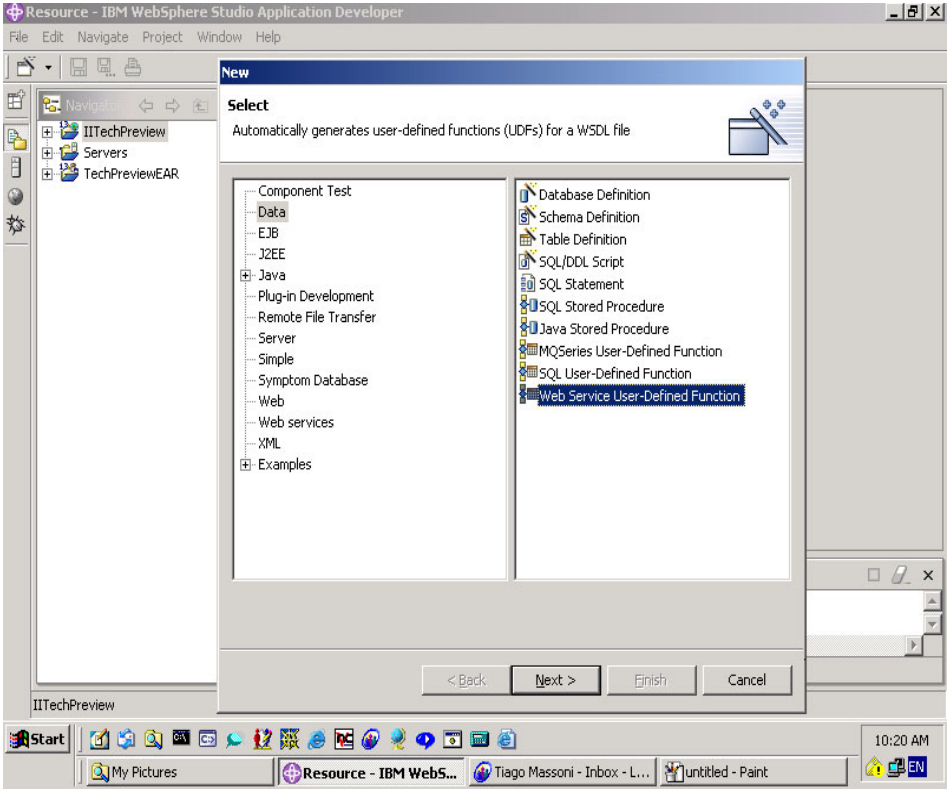


Figure 5-22 Web Service query generation with WSAD

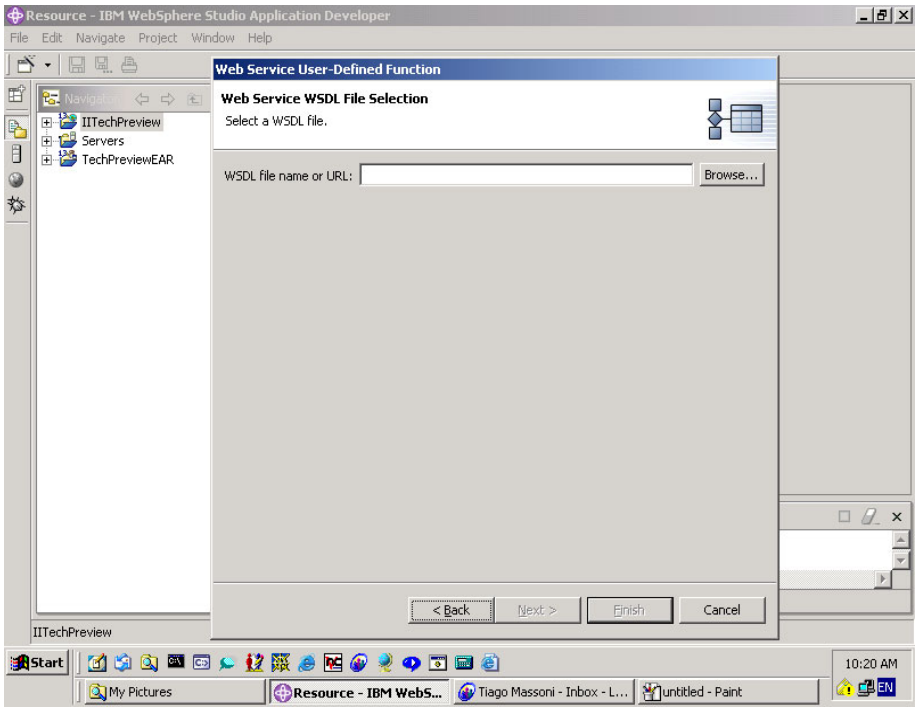


Figure 5-23 Web Services query definition through WSAD

- A “view” is constructed and associated with each of the data sources described in the high-level unified information view shown in Figure 5-19 on page 170. This view is constructed for the data source regardless of whether it is a Web Service, XML, or relational data source. The view describes the information to be obtained from the source, and also any transformations to the data that may be required.

  The XML Extender function in DB2 II is used to decompose the information coming from Supplier Y into DB2 XML collections in the data federation server. This is done using the XML Extender Administration tool.

- The “federated” view is then constructed as a UNION of all the views from the individual sources. This allows a single SQL query to be submitted to the federated system as if it were a single data repository. The federated system obtains current information in real time from the operational data sources. This information is then represented in XML using DB2 XML extender in the federation server. The results are represented as XML collections.

For further details on the toolset, refer to the IBM Redbook, *Getting Started on Integrating Your Information,* SG24-6892, and the demonstration on DB2 II at:

`http://www.ibm.com/developerworks`

Displaying the information through a portal

Portals get information from local or remote data sources, for example, from databases, transaction systems, content providers, or remote Web sites. They render and aggregate this information into a compact and easily consumable form. The data retrieved by the portlet is usually local but could also be remote, retrieved possibly using Web Services.

As these Web Services are becoming the predominant method for making information and applications available programmatically through the Internet, portals need to allow for integration of Web Services not only as simple data sources, but also as remote application components. There are three important options for using Web Services in conjunction with portals:

- Portlets running on a portal server access Web Services to obtain information or invoke remote methods provided by the Web Service.
- Content or application providers can implement and publish interactive, user-facing Web Services that plug and play with portals.
- Portals can publish local portlets as remote portlet Web Services to make them available to other portals.

IBM WebSphere Portal supports all of these options.

In the ABC scenario, we implemented the first option. The Web Service gets informations for an item, but then the portlet has to provide the user interface. This decoupling between the presentation of the data and the query is a good thing. It will allow applications other than the portal to use the Web Service query.

Business objective

In the last section, ABC created a simple Web Service to query information on the current inventory level, the price, and the delivery dates of a specific item. Now it would be a good idea to give the ABC analysts and the reseller partners the ability to see that information as well.

The portal allows the data to be displayed in a customized manner, based on the user profile. Indeed, the presentation of the information is not the same for an analyst or a reseller partner.

In this section we describe deploying a portlet that formats the data returned from the data-oriented Web Service.

A data-oriented Web Service is a Web Service that returns only unformatted data. In this case, it simply returns raw XML.

The purpose of this method of using Web Services and portlets together is to utilize available services while maintaining control of how an individual program uses and displays the information.

Architecture of the portal application

There are, of course, many ways to set up a portlet to access a Web Service and retrieve the list of information on the items to display to the users. The Web Service could be called directly from the portlet, the formatting performed by the service, and HTML returned to be displayed. But really, what fun would that be? Instead, we can build a slightly more scalable and well-balanced architecture.

Most importantly, a few components that have their own special jobs to perform will be created. If these “jobs” are combined into a single component, we run the risk of limiting reusability and scalability. The call to the Web Service is moved out of the portlet to another bean that will handle that task. It's possible that some other portlet might also want to access the Web Service. For that matter, it might not be a portlet at all that wants to access the Web Service, but rather something completely different like a servlet, fat client, or another Web Service. The portlet we are creating has one purpose: to provide the portlet-style presentation of the application. All of the logic happens in the command bean. The command bean's methods are simply invoked from the portlet and then control is passed to one of two JSPs (depending on whether the portlet is maximized or minimized).

In addition, there are a couple of other JavaBeans used to store data from the XML file returned from the Web Service. A method takes the XML document returned from the Web Service and strips out the data, storing it as various properties of a data bean. For each piece of information on an item that is returned, there will be a separate data bean instantiated and the collection of those beans will be made available to the JSP code. Beans are simply easier to handle than XML from a JSP file.

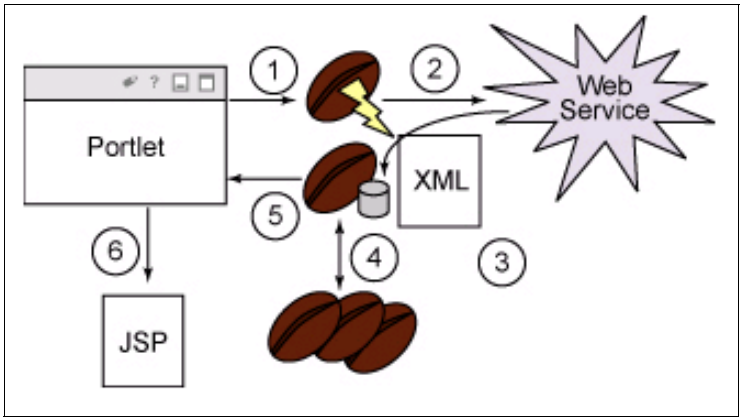


Figure 5-24 Portal Application Flow

The portal application flow illustrated in Figure 5-24 occurs as follows:

1. The portlet instantiates the command bean and calls the setDoc() method.
2. The method setDoc() on the command bean calls the Web Service and sets the property doc to the XML returned by the Web Service.
3. The XML stored in doc is passed to the collection bean.
4. For every item node of the XML, a data bean is instantiated and properties of it are set based on the values held within the XML.
5. The entire collection of item data beans is stored in the portlet request.
6. Control is passed to the JSP that uses the item collection stored in the request to display the list of all the information on the items in an HTML table format.

Enabling user interaction

ABC will use the portal server to provide a personalized, consistent user interface to its internal employees, as well as to business partners that access its systems. The portal provides dynamic access to the following functions:

- Any collaborative functions required by team members on a particular function or project.
- Consistent access to line of business applications, such as wholesale, retail, and sales. Some of these capabilities are enabled through modernization of the interface of existing legacy applications through tools such as HATS. Detailed discussion of these is outside the current scope of this book.
- Access to other common intranet applications, such as HR, internal directory, and so on.

- Managed access to other sources of domain information on the electronics industry, such as breaking news stories and press releases.

These functions are created as “portlets” which can be included as tabs in a user’s personalized interface. The functions that a user can access are managed based on profiles, such as line of business and role, which also governs access privileges. Within the scope of this profile the user can personalize and customize the interface.

For the users in the wholesale department, including the inventory analysts, a collaborative TeamSpace is also provided so that they can obtain rapid resolution to inventory issues. The following figures are illustrative of the types of collaboration and interaction the users at ABC are enabled for.

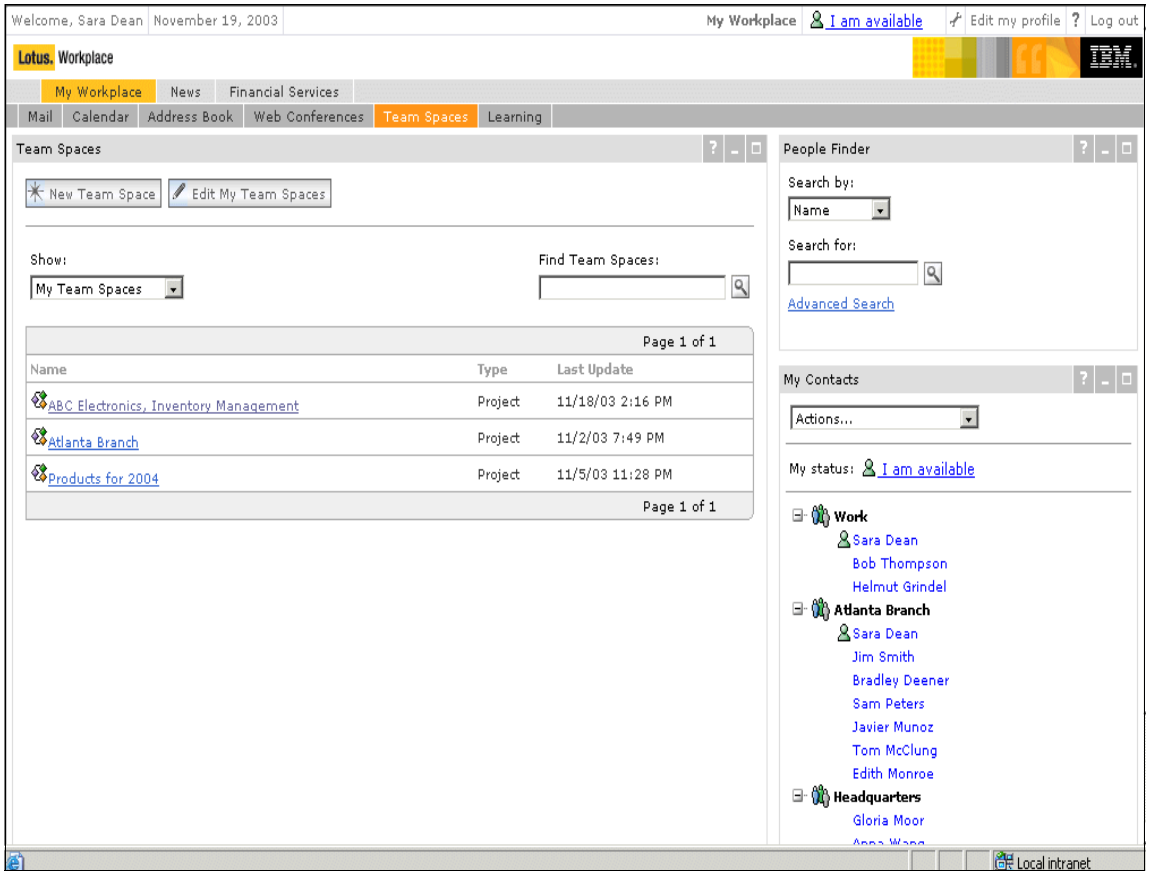


Figure 5-25 ABC Electronics - TeamSpaces

Figure 5-25 shows the team spaces set up for various teams in ABC Electronics.

There are three TeamSpaces available at ABC currently:

- Inventory management
- Atlanta branch
- Marketing focused TeamSpace dealing with products for 2004.

The column on the right-hand side shows the various participants that can participate in the TeamSpace. There is also the capability for adding users based on proper authorization.

Team members from ABC's wholesale function collaborate in the inventory management TeamSpace to resolve issues related to ABC's inventory.

The inventory analysts select the inventory management TeamSpace, a sample of which is shown in Figure 5-26.

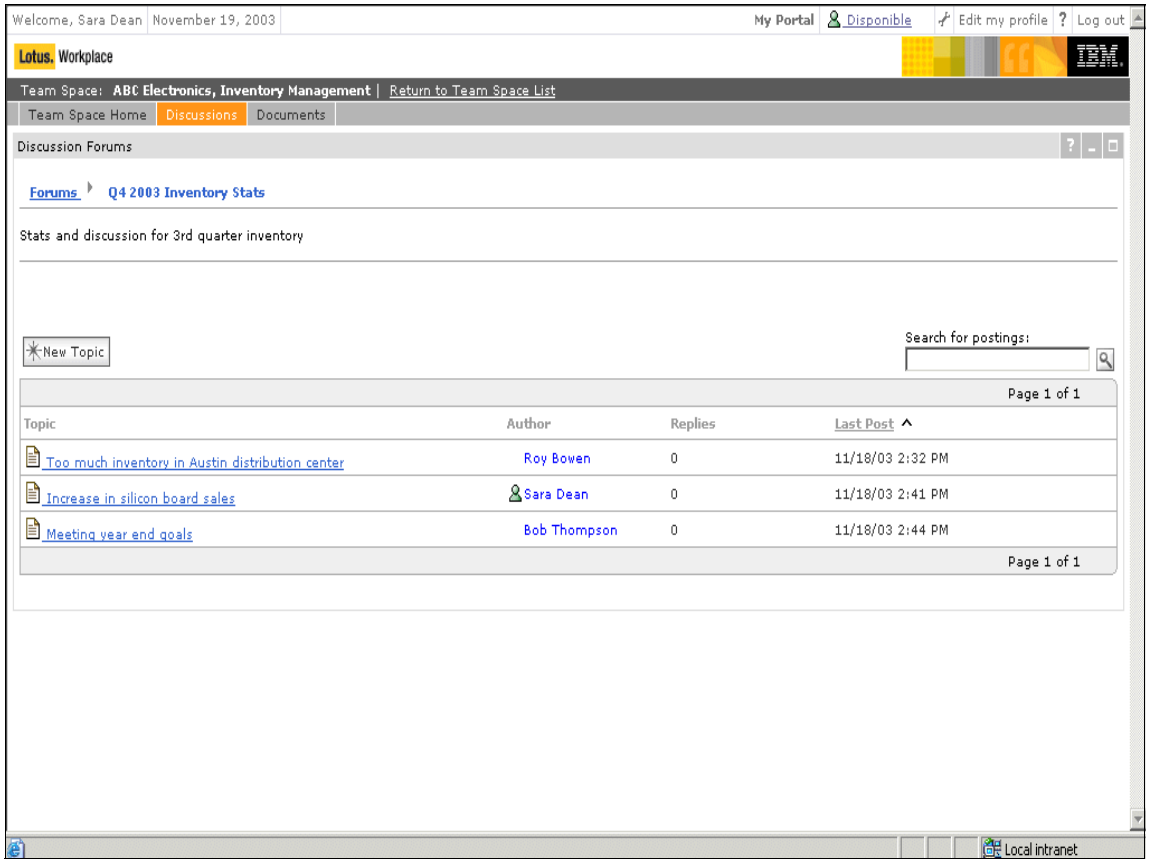


Figure 5-26 Discussion threads in the inventory management TeamSpace

There are discussion threads on inventory levels in a distribution center, sales of particular items, and business objectives. The analysts can collaborate on individual projects or issues through these discussion groups as well as through other techniques such as instant messaging to achieve their project goals.

In addition to discussion groups and instant messaging, the groups can refer to documents related to their project domain, as shown in Figure 5-27.

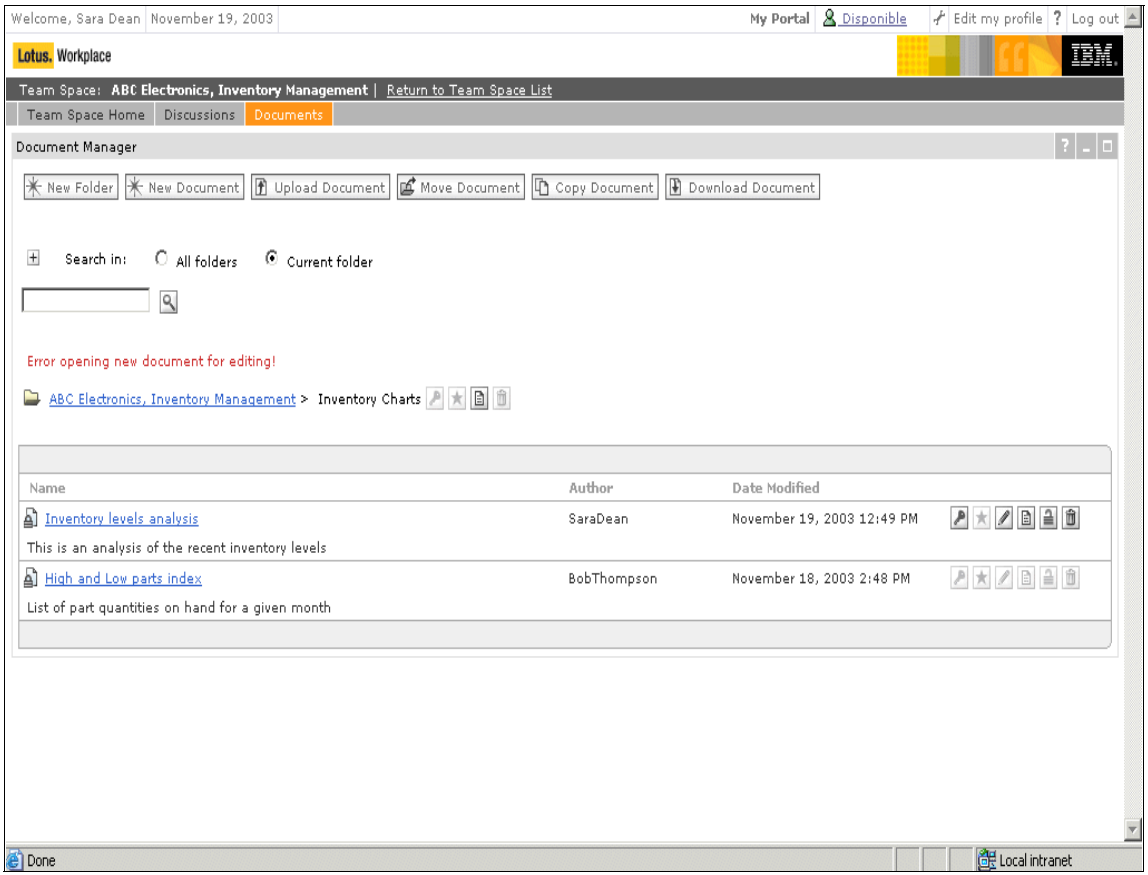


Figure 5-27 Documents in the inventory management TeamSpace

These capabilities allow the inventory analysts to obtain rapid resolution to any business issues and allow them to dynamically set up collaborative environments as project conditions require. This mirrors the ad hoc nature of project and cross-functional teaming arrangements required in a flexible business organization.

Infrastructure updates

Since ABC has WebSphere Application Server installed and they have developed J2EE and Java programming skills, they are familiar with the WebSphere Application Developer Studio - Integration Edition.

At this point the IT department needs to determine an IT infrastructure architecture that will be implemented to support the information requirements. In order to do this, there are several guidelines the IT department has established, such as:

- The Web is a preferred channel for interaction with users, both internal and external.
- Leverage investment in existing applications to minimize risk and lower costs of development.
- Develop a standards-based infrastructure that is flexible and can communicate with a wide variety of systems, both internal ones and external systems such as those of suppliers, customers, and partners.

Based on these guidelines, the IT department at ABC Electronics evaluated various alternatives and developed the high-level conceptual architecture shown in Figure 5-28.

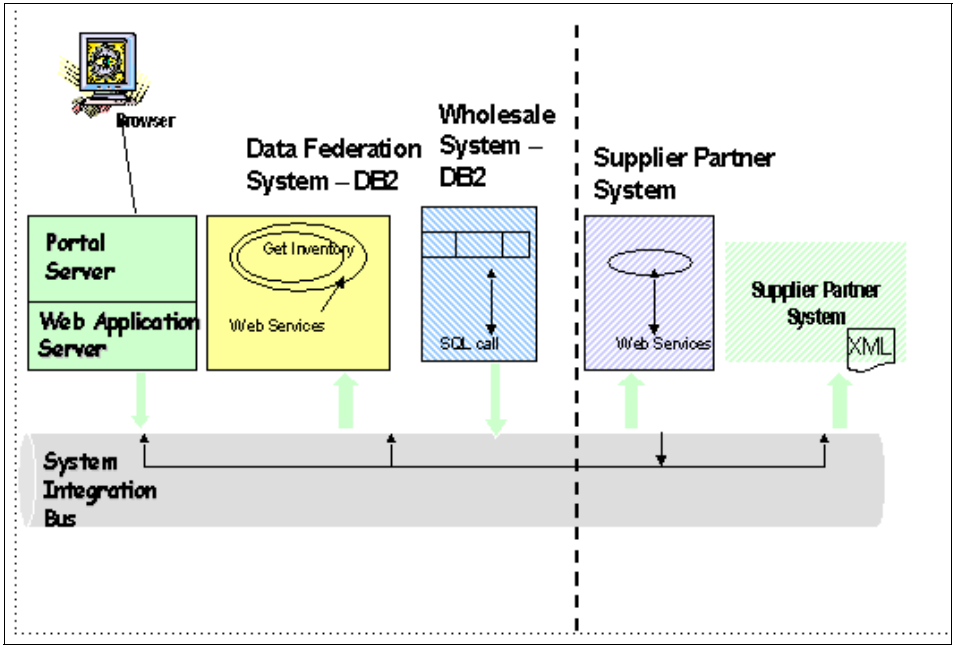


Figure 5-28 Conceptual integration architecture

As described in 4.4.2, “ABC Electronics scenario” on page 116, ABC decided to adopt a Services Oriented Architecture (SOA) for their information integration needs. The heart of the SOA is the Enterprise Service Bus (ESB) as shown in Figure 5-28. This approach allows ABC to:

- Leverage core legacy systems and communications between them.
- Leverage close integration between WebSphere and DB2 II. This enhances developer productivity through consistent use of toolsets such as WSAD.
- Envelope information from various sources into Web Services. These Web Services are exposed via an Enterprise Service Bus to various consumers of Web Services.
- These service components are aggregated at a coarse-grained level to support new business processes, promote integration, and interoperability with other systems.
- The service components themselves may be assembled and aggregated to promote reuse and provide rapid time to value for any new products or processes being considered. Typically, guidelines are that no more than five components (representing transactions) should be aggregated to expose a “service.”
- The ESB provides common system services such as access control and data caching.
- The following additional components are installed and connected to the ESB by ABC:
  - Caching is deployed at various levels in the infrastructure to improve service levels.

    Data caching provided by DB2 II allows frequently accessed data from various sources to be accessed without having to go to the various data sources. The caching can be configured to ensure that time-sensitive data is refreshed as required.

    The DB2 II data cache is also used to enhance availability. Materialized Query Tables (MQT) are set up at the federation server to provide a fallback in case one of the data sources is not available, thereby preventing the federated query to the sources from failing.
  - WebSphere Portal Server and Lotus Workplace are installed to enable user interaction as discussed earlier.
  - WORF and DADX components of WebSphere Application Server EE are utilized. These are included with DB2 as described earlier, and allow Web Services requests to be fulfilled by DB2 II.

A functional layout is shown in Figure 5-29.

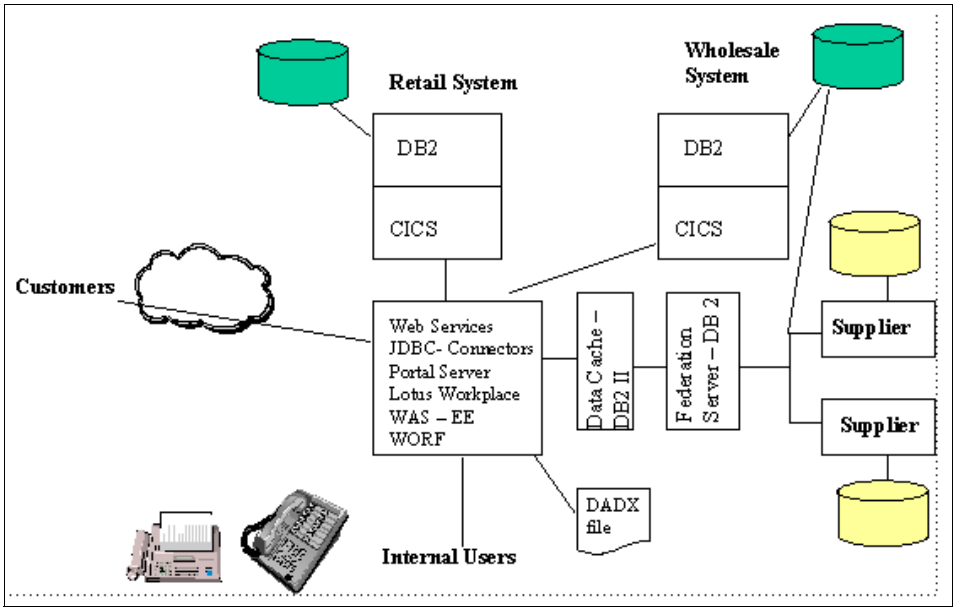


Figure 5-29 Proposed functional layout

Since information is being accessed from outside the enterprise boundaries, special attention needs to be paid to security and access control relative to the value of the information being protected. In this case, ABC has taken the following measures to ensure that access is controlled consistent with their policies:

- No public Internet access to business partner systems. VPNs are used.
- Query only capability to start with. No write access from supplier systems.
- Access to the supplier systems is by authorized and trusted users only through the user mapping function described earlier. The user credentials are passed to the partner system through the wrapper or the Web Service.
- Appropriate auditing and control mechanisms are in place to monitor access and make sure any exposures are identified and handled appropriately.

The SOA described in this section provides a foundation for future enhancements of the information integration capability at ABC. For example, in the future this provides the basis for direct ordering by reseller partners through ABC to supplier partners, allowing ABC to be an intermediate to the sale without the expense of maintaining inventory. In order to provide such capabilities, additional transactional components will need to be considered for the infrastructure in order to ensure integrity of order flow and information.

5.4.6 Benefits and summary

In this simple scenario, we saw how a hypothetical retail enterprise was able to rapidly leverage the information available in their enterprise and with partners to improve their business efficiency and provide an additional service to their customers. Among the key benefits obtained by ABC were the ability to:

- Rapidly integrate diverse sources of information to provide additional value to customers, as well as higher efficiency through enhanced inventory management.
- Leverage its investment in existing systems. The approach discussed in this section was an evolutionary approach, which did not involve “rip and replace.” The investment in the Retail and Wholesale systems was preserved.
- Leverage their current skill set. ABC leveraged their existing J2EE skill sets through WSAD and WAS -EE, and DB2 skills, as well as synergy between the toolset as described earlier.
- Leverage the federation capabilities of DB2 - II to provide a unified view of information in the company.
- Reduce the risk of errors since there is less code to implement, and achieve more rapid time to value.
- Leverage the synergy between technology components, especially WebSphere and DB2, to expose the unified view as Web Services for other processes to use. This allows flexibility for further integration with other processes in the future.

Some considerations to note:

- Information integration requires the source systems to be available at the time of query. This requirement is reduced through the use of DB2 cache as an alternate store in case of non-availability of the original data source.
- Techniques for loose coupling of external systems described briefly in Chapter 2 should be considered by ABC for future enhancement.
- As ABC moves to more fully integrate ordering from reseller partners to supplier partners in future, they may need to consider integrating transactional capability to the information integrating capability. One option to provide this could be through the integration between DB2 II and WBI Integrator that is being developed.

5.5 Product positioning

IBM DB2 Information Integrator V8.1 provides integrated, real-time access to diverse data as if it were a single database, regardless of where it resides.

The federated server lets users:

- Create an abstract relational view across diverse data.
- Use existing reporting and development tools.
- Rely on leading-edge cost-based optimization.

The replication server lets users:

- Manage data movement strategies including distribution and consolidation models.
- Monitor synchronization processes.

Supported data sources include, among others:

- DB2 Universal Database
- Informix®
- MS SQL Server
- Oracle
- Sybase
- Teradata
- ODBC

Supported content sources include, among others:

- WebSphere MQ message queues
- Lotus Notes
- Documentum Enterprise Content Management System
- Web search engines and Web Services
- MS Excel spreadsheets
- XML docs

IBM DB2 Information Integrator for Content (formerly Enterprise Information Portal in versions 8.1 and earlier) provides broad information integration and access to:

- Unstructured digital content such as text, XML and HTML files, document images, computer output, audio, and video
- Structured enterprise information via connectors to relational databases
- Lotus Notes Domino databases and popular Web search engines using IBM Lotus Extended Search
- Objects within business process workflows

By using DB2 Information Integrator for Content V8.2 that leverages the power of IBM WebSphere MQ Workflow, and IBM Lotus Extended Search, users can personalize data queries and search extensively for very specific needs across traditional and multimedia data sources. Developers can more rapidly develop

and deploy portal applications with the information integration application development toolkit.

Table 5-1 describes the product positioning between DB2 II and DB2 II for Content.

Table 5-1 Product positioning

DB2 Information Integrator	DB2 Information Integrator for Content
SQL programming model.	Content programming model.
Leverage SQL skills and tools.	Leverage CM skills and tools.
Federated data server and replication server.	Federated data server, text mining, and workflow engine.
Majority of the information is in relational data stores.	Majority of the information is in DB2 Content Manager and unstructured content stores.
Relational join or union capability is needed.	Relational join capability is not required or is handled by some other solution component.
Application is primarily decision support. Limited update to RDBMS source is provided; update to content sources is not provided.	Application is primarily decision support. Read/write access is available for some content sources but is not the primary requirement.
Query results can be made available to WebSphere MQ message queues or published as XML documents.	Application may involve other activities such as integrating with document workflow or information mining.

IBM DB2 Information Integrator and DB2 Information Integrator for Content both offer federation access diverse and distributed data and content stores, but each presents a different programming model tailored to a different developer community. The decision of which one to use depends on needs as identified with the following questions:

1. Which types of data and /or content sources need to be federated?

2. Are the application developers who will work with the federated application more comfortable with SQL as a programming language or are they really more content management/ECM-centric and want to federate their content stores?

Both products can be integrated with IBM Lotus Extended Search to broaden the range of the content sources, so the “programming model” question is very important. These products provide the strategic foundation for a framework that helps customers access, manipulate, and integrate diverse, distributed and

real-time data. Organizations may want to use one or both products, depending on the style of programming desired, and the relative mix of content versus data to be accessed.

DB2 Information Integrator is well suited for IT organizations in which development expertise is concentrated in structured query language (SQL), extensible markup language (XML), or other RDBMS-centric applications and tools. For these businesses, the primary data sources are relational data sources augmented by other XML, Web, or content sources.

Where developers are focused on content issues and inclined to write application code against the same content management application programming interface (API) as their enterprise content management (ECM) solution, DB2 Information Integrator for Content is preferred. DB2 Information Integrator for Content supports a variety of data and content sources including: content repositories such as the IBM DB2 Content Manager family and FileNET, relational databases such as IBM DB2 Universal Database, Oracle, Lotus and Microsoft e-mail systems, and information across file systems and the Web.

5.6 Linkages

The following sections describe linkages between some of the key products that have been discussed in this chapter. Many of these linkages are dependent upon standards such as XML.

5.6.1 XML

XML is simple, open, universal, extensible, flexible, and controlled. These terms are only a subset of the possible reasoning or justification in answer to the question: Why XML? In fact, XML (eXtensible Markup Language), a W3C standard, is an accepted and viable interface option for many business integration scenarios. XML is a simple text-based notation and tagged message format that enables loose coupling of sending and receiving applications.

XML imposes a measure of control by requiring the presence and proper nesting of user-specified tags, allowing for well-formed XML documents. Control is also manifested by optional use of DTD (Document Type Definition) or XML Schema validation mechanisms for checking XML documents with an XML parser.

XML is also a notation for defining tagged languages. This capability allows XML to be widely used and extended by many businesses to create a customized vocabulary for communication with customers and business partners. In essence, one is able to create a unique language for communication. This is due,

in part, to the device-independent and platform-independent nature of XML in facilitating communication and information sharing between disparate systems.

Note: Additionally, XML documents can be accessed via the XML wrapper in DB2 Information Integrator.

Though XML, by itself, is an interesting and useful technology, the combination of XML and database technologies provides a more powerful and persuasive argument for using XML. Indeed, there are some computing problems better suited to the use of a database management system. Enabling quick information searching and providing a central repository for storing XML documents are only two of the possible capabilities made available by using a relational database management system.

XML support in DB2 UDB is provided by a set of SQL functions, data types, and stored procedures that manipulate XML documents. In general, the stored procedures are used for composition and decomposition. However, a large number of user-defined functions (UDFs) are used for storing intact XML and integration with the file system and WebSphere MQ. In addition, XML Extender provides a set of new data types that are also used for direct storage of intact XML.

User-written DB2 UDB client programs can use the SQL, SQLJ, ODBC and JDBC application program interfaces (APIs) to access the IBM supplied or user-defined functions and stored procedures to exploit XML. DB2 XML applications can be Java servlets, stored procedures, user-defined functions, PC clients, CICS, and IMS, among others. The application can be written in Java, C, C++, COBOL, REXX, Perl, SQL stored procedures, and many other languages capable of using the DB2 UDB database APIs. An XML Parser, installed separately and used by DB2 UDB, is also shipped with XML Extender.

DB2 Information Integrator contains both DB2 UDB and the DB2 XML Extender. While the term DB2 is used in the following discussion, the XML capabilities are also in DB2 Information Integrator.

5.6.2 DB2 XML Extender

DB2 XML Extender provides the stored procedures, user-defined functions and user-defined data types to store, compose, shred, validate, and search XML documents in a DB2 UDB database or on a local file system. See Figure 5-30 for a high-level depiction of the DB2 XML Extender components.

Decomposition (shredding): This is the process of storing an XML document in DB2 UDB. The XML document is broken apart (or shredded) and the elements are stored as fields in one or more DB2 UDB tables.

Composition: This is the process of creating (or composing) an XML document from data in DB2 UDB tables. Elements in the generated XML document are created from fields in one or more DB2 UDB tables. The XML document can be stored in DB2 UDB or outside of DB2 UDB, in the file system and WebSphere MQ message queues.

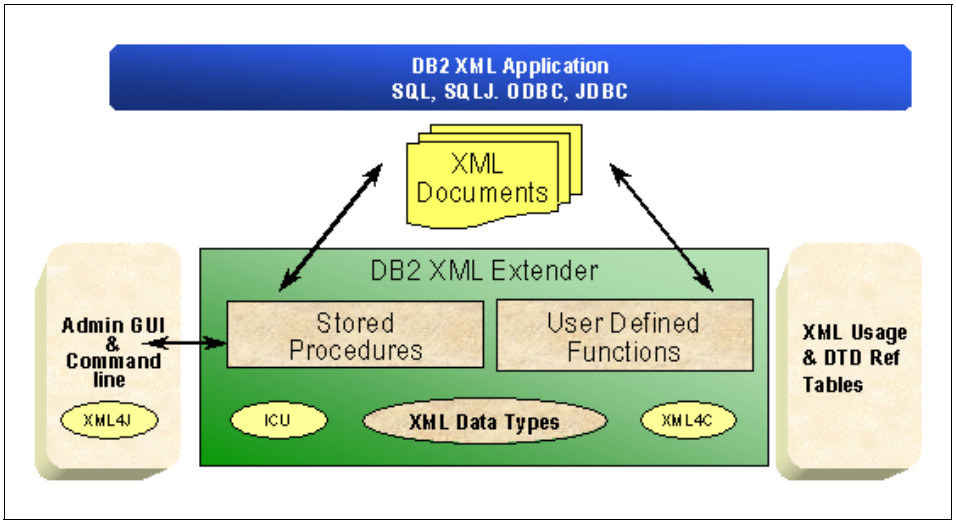


Figure 5-30 XML Extender components

XML Extender is capable of importing or exporting XML documents that are in memory, in the file system, or in WebSphere MQ queues. In addition, XML documents can be transformed by using XML style sheets (XSL). The overview diagram in Figure 5-31 is used to show how XML data sources are integrated with DB2 XML Extender.

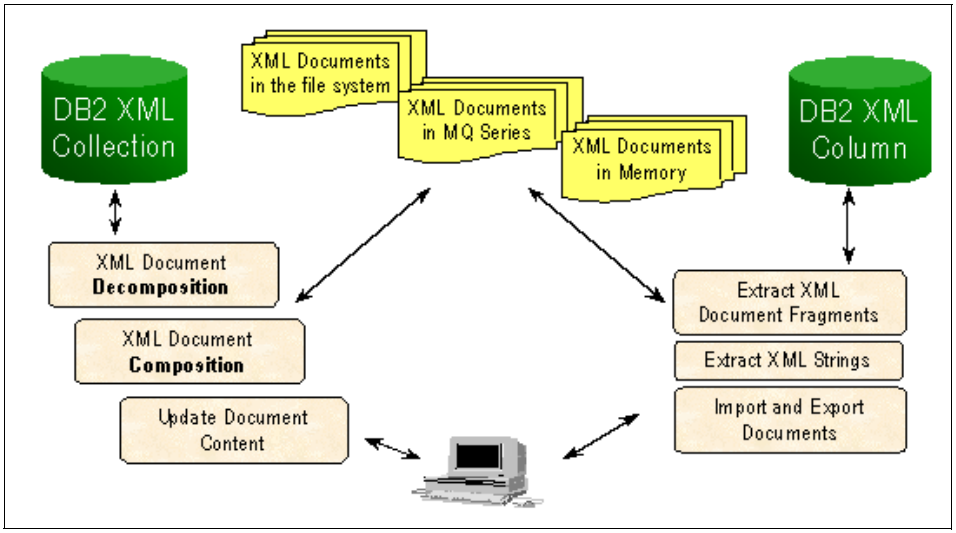


Figure 5-31 XML data source integration with DB2 UDB

Document Access Definitions (DADs) are XML documents used for mapping XML and relational data; and DTD files are stored in the XML Extender tables, XML_USAGE and DTD_REF. The contents of DAD files determine whether an XML column is used to store an XML document in DB2 UDB or an XML collection is used to store or generate an XML document to/from DB2 UDB tables. DTDs are used to validate the structure of XML documents.

XML Extender also uses a subset of the XML Path Language (XPath) to navigate the structure of an XML document. XPath is used by XML Extender functions to identify elements and attributes when extracting or updating XML columns. In regard to XML collections, XML Extender stored procedures use XPath to override values in the DAD file. Example 5-1 shows two examples of specifying an XPath.

Example 5-1 Specifying an XPath

```
/PersonnelRec/Person/Name/Family
/PersonnelRec/Person/@Salary
```

Figure 5-32 highlights the storage options available in DB2 XML Extender.

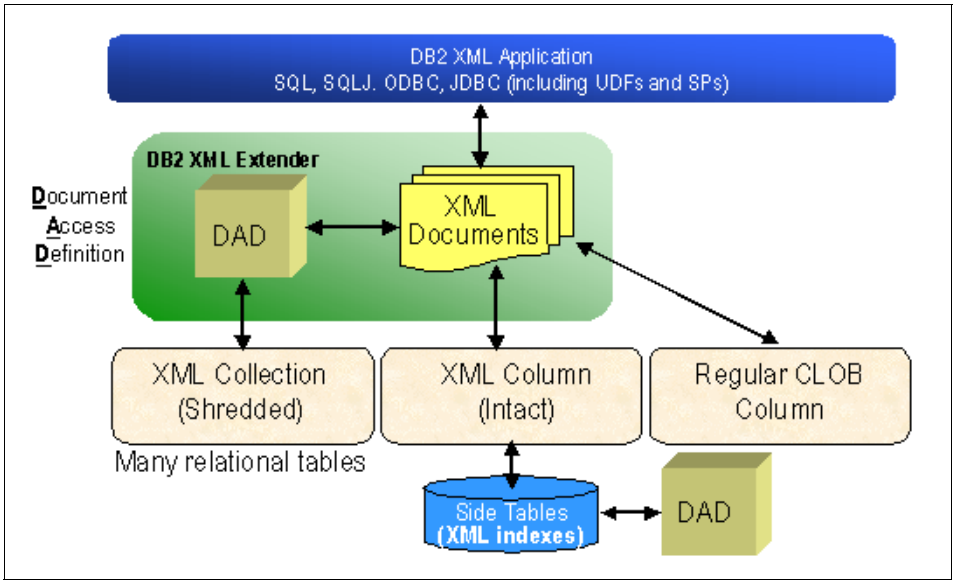


Figure 5-32 XML storage options in DB2 UDB

XML Extender provides the following three storage options for storing XML documents in DB2:

- **XML columns:** An XML document is stored, as-is, in an XML-enabled column of a database table. This XML column can be updated, retrieved, and searched. In addition, the element and attribute data of an XML document can be mapped to DB2 UDB side tables and indexed to exploit the RDBMS search capabilities available in DB2 UDB. Following are some of the reasons that one might choose to use an XML column instead of an XML collection:
  - The XML documents already exist.
  - You need to store intact XML documents for archiving or audit purposes.
  - Frequently searched XML elements and attributes are known in advance.
  - Frequently read and rarely updated XML documents are known in advance.
  - You need to keep XML documents external to DB2 UDB on a file system.

  XML Extender provides the XMLVarchar, XMLCLOB, and XMLFile data types for direct storage of XML data in an XML column. Of the three new data types, XMLCLOB is most frequently used, due to the maximum size limit of two gigabytes. The XMLVarchar data type is used less because the three kilobyte maximum size creates more uncertainty as application designers

cannot guarantee that the size of XML documents will be within the three kilobyte maximum. The XMLFile data type is used to store and manage XML documents externally (outside of the DB2 UDB system) on the file system.

Although the DAD file maps XML to relational data, it is primarily used to configure indexing (optional) of data in an XML column. In addition, the DAD file specifies a DTD file for validating XML documents stored in an XML column.

After an XML document has been stored in an XML column, the user has the ability to extract, update, and search XML elements and attributes. The <Xcolumn> tag is used to specify that Xcolumn access and storage is required.

- **XML collection:** XML collections enable the mapping of XML document structures to DB2 UDB tables, and facilitate the storing of XML documents in DB2 UDB or the creation of XML documents, outside of DB2 UDB, from content residing in DB2 UDB tables. Basically, an XML document is shredded (or decomposed) and the untagged pieces of data are stored in the columns of one or more DB2 UDB tables. Conversely, an XML document can also be generated (or composed) from data existing in DB2 UDB tables. Following are some of the reasons that one might choose to use an XML collection instead of an XML column:
  - Performance of updates is critical.
  - There may be a need to:
    - Generate XML documents from DB2 UDB tables.
    - Generate XML documents from specific columns.
    - Store untagged portions of XML documents for other application access.
    - Drive existing business systems using the document content.
    - Frequently update small parts of an XML document.
    - Process document content with analytical software.

  We recommend that XML documents be validated by a DTD file, stored in the DTD repository, prior to shredding in DB2 UDB tables. It should be noted that, for composition, only one DTD can be used to validate generated XML documents. Multiple DTDs can be specified in a DAD file, however, when decomposing XML documents.

  The <Xcollection> tag is used to specify that Xcollection access and storage is required.

  When using XML collections, a mapping scheme for XML and relational data must be selected and configured in a DAD file. XML Extender provides the SQL statement and RDB node mapping schemes for defining an XML collection in a DAD file.

- **SQL mapping:** This scheme uses a SQL SELECT statement to define the DB2 UDB tables and conditions used for composing an XML document. The SQL mapping scheme enables a direct mapping of relational data to an XML document using a single SQL statement and the XPath data model. The SQL mapping notation is broken into two parts. The first part consists of a SQL statement that provides the scope (rows and columns) of the relational data to be included in the document. The second part, beginning with the root node, describes the shape of the XML document. It should be noted that the SQL mapping scheme is a one-way notation and cannot be used for decomposition.
- **RDB node mapping:** This scheme uses a two-way mapping notation that employs an XPath-based relational database node (RDB) to compose or decompose an XML document. The DAD file RDB nodes contain definitions for tables, optional columns, and conditions that assist XML Extender in determining where to store or retrieve XML data. The RDB node mapping notation is also broken into two parts. The first part is used for scoping. However, there is no SQL statement as is the case for the SQL mapping scheme. Scoping is accomplished by listing the required relational tables, along with their relationships, that one wants to appear in the XML document. The second part is used for shaping and describes the mapping of relational data to the element and attribute names in the order and nesting level as needed for appearance in the XML document.

- **Regular CLOB (Character Large OBject) column:** XML Extender provides the XMLCLOB UDT, created from the existing CLOB data type, to support XML data in DB2 UDB. XML documents can be stored, however, as a regular CLOB data type.

5.6.3 DB2 XML Extender and WebSphere MQ

The WebSphere MQ user-defined functions available in XML Extender are intended to provide database programmers and administrators with an introduction to MQSeries XML message integration with the WebSphere MQ family (including WebSphere MQ, MQSeries Publish/Subscribe and WebSphere MQ Integrator).

These functions are used to pass only XML messages between DB2 UDB and the various WebSphere MQ implementations. Use of DB2 Information Integrator extends the passing of XML messages to non-DB2 sources and the various WebSphere MQ implementations.

In addition, the MQSeries XML Extender functions must be enabled for access in DB2 UDB. This is done with the following Enable_MQXML command:

```
enable_MQXML -n DATABASE -u USER -p PASSWORD -
```

All MQSeries XML functions have a DB2XML database schema and are listed in Table 5-2. XML Extender uses XPath and SQL to manipulate XML documents.

Table 5-2 MQSeries XML functions

MQSeries XML Function	Description
DB2MQ.MQSendXML	Send an XML message to the queue.
DB2MQ.MQReadXML	A nondestructive read of matching XML messages from the queue.
DB2MQ.MQReadXMLAll	A nondestructive read of all XML messages from the queue.
DB2MQ.MQReadXMLCLOB	A nondestructive read of matching XML CLOB messages from the queue.
DB2MQ.MQReadXMLAllCLOB	A nondestructive read of all XML CLOB messages from the queue.
DB2MQ.MQReceiveXML	A destructive read of matching XML messages from the queue.
DB2MQ.MQReceiveXMLAll	A destructive read of all XML messages from the queue.
DB2MQ.MQReceiveXMLCLOB	A destructive read of matching XML CLOB messages from the queue.
DB2MQ.MQReceiveXMLAllCLOB	A destructive read of all XML CLOB messages from the queue.

In addition, the MQSeries text functions must be enabled for access in DB2 UDB and DB2 Information Integrator. Furthermore, the MQSeries text functions have a DB2MQ database schema and are not considered to be part of the XML Extender DB2XML database schema. Following is an example of the Enable_MQFunctions command:

```
enable_MQFunctions - DATABASE -u USER -p PASSWORD
```

IBM provides the ability to store and retrieve not only structured relational data, but also semi-structured XML documents and unstructured content, such as byte streams and scanned images. XML documents can be loaded in DB2 UDB (and DB2 XML Extender used to manipulate them) or they can be manipulated using new DB2 SQL/XML functions. They can also be accessed in place using the XML wrapper provided by DB2 Information Integrator. With the content of XML documents in a DB2 UDB database, one can combine XML information with traditional relational data. Based on the application, one can choose whether to store entire XML documents in DB2 UDB as in user-defined types provided for XML data, or map the XML content as base data types in relational tables.

Publishing and interchange through XML documents is a key technology that enables an enterprise to share data with internal and external applications in a common format. In addition, information integration provides a facility for the interchange of XML data and to manipulate XML data. It includes:

- The capability to store entire XML documents as column data or externally as a file and extracting the required XML element or attribute values and storing them in side tables, indexed sub-tables for high-speed searching. Storing XML documents as column data is known as the XML column method of storage and access. By storing the documents as column data, one can:
  - Perform fast search on XML elements or attributes that have been extracted and stored in side tables as SQL basic data types and indexed.
  - Update the content of an XML element or the value of an XML attribute.
  - Extract XML elements or attributes dynamically using SQL queries
  - Validate XML documents during insertion and update.
  - Perform structural-text search with the text extender.
- The capability to compose or decompose contents of XML documents with one or more relational tables, using the XML collection method of storage and access. XML documents can be composed from relational storage and also WebSphere MQ and file system data stores.

5.6.4 Integration of applications using WebSphere MQ products

Key to developing a corporate integration infrastructure is the ability to easily leverage appropriate integration technologies, together or separately. The information integration infrastructure provides client APIs that leverage messaging and workflow facilities, such as those provided by the WebSphere software platform from IBM. Thus, a database event—for example, the arrival of a new piece of information—can transparently create a notification by putting a new message on a queue.

This allows information integration to facilitate the full participation of internal and external applications in the business processes of a given enterprise. This can be accomplished using the MQSeries Workflow and WebSphere MQ products.

Message queuing is a data transport mechanism that is often described as “e-mail between applications.” A sending or requesting application formats a message, a structure that can contain any type of data, and writes the message to a data store called a queue. The serving or receiving application then retrieves the message from the queue and processes the data. This is a simple concept that provides effective decoupling of applications, allowing an application to communicate information to another without being directly connected. The sender and receiver do not have to reside on the same platform; in fact, neither

the sender nor receiver has to know where the other application is hosted. Nor do they have to be running at the same time. Messages will remain queued until they are retrieved by an application or expire.

WebSphere MQ supports two messaging styles used by information integration: datagrams and request/reply messaging. Datagrams, also known as “fire and forget,” are used to send or publish information without expecting a response from the receiving or subscribing application, and are often used in data replication. Request/reply messages provide communication between requesting applications and the servers that fulfill the request. This style is often used in providing business services to both internal and external clients.

While WebSphere MQ is basically an asynchronous communication method, many request/reply applications actually run in what we call “near real time.” That is, the initiating application expects a reply from the service request. Please note that this does not imply that the application waits on a reply, although it frequently does.

In addition, WebSphere MQ runs on 35+ platforms and uses a common API on all these platforms. More importantly, WebSphere MQ guarantees persistent message delivery.

For example, DB2 Information Integrator enhances the WebSphere MQ integration in DB2 UDB with access to federated heterogeneous data.

With MQSeries Integration, DB2 UDB stored procedures and user-defined functions and triggers can send and receive text or XML messages to/from WebSphere MQ queues. In addition, WebSphere MQ queues can be used as a data source for SQL calls. Message queuing facilitates data interchange with traditional business integration implementations. A DB2 UDB developer or administrator can use SQL statements and does not have to learn the WebSphere MQ Integrator to access WebSphere MQ queue data.

5.6.5 Web Services integration

Figure 5-33 illustrates how DB2 UDB and DB2 Information Integrator will utilize Web Services to:

- Provide access to data that is stored inside DB2 UDB to additional clients. In this case, they are the SOAP clients (see the right-hand side of Figure 5-33).
- Act as a consumer of Web Services to bring external data and serve the data as part of the data stored in DB2 UDB (see the left-hand side of Figure 5-33).

In this discussion, we only concentrate on examining DB2 UDB acting as a Web Services provider.

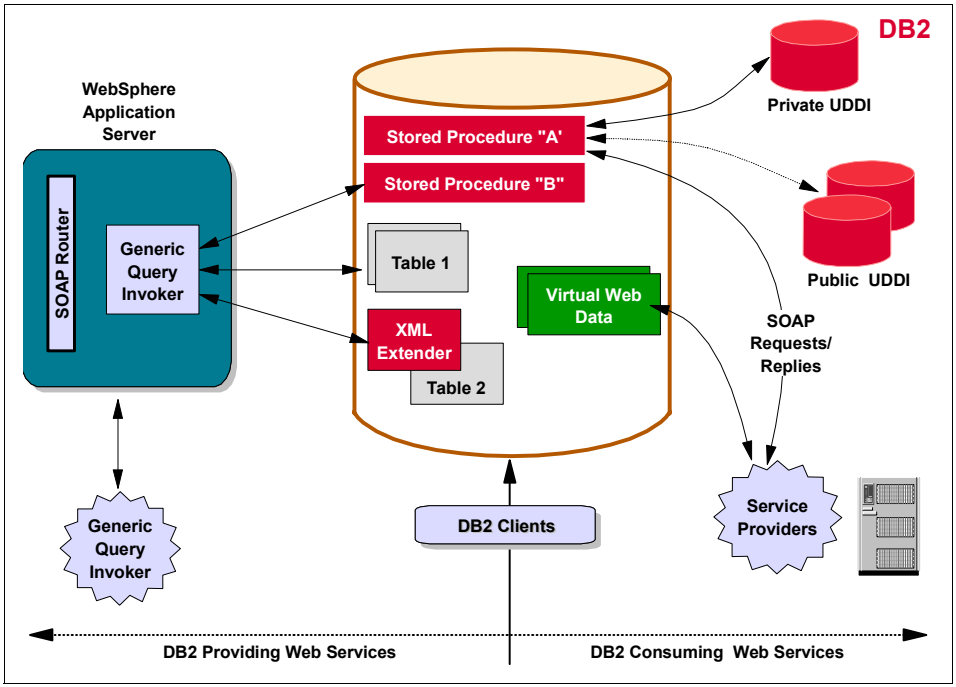


Figure 5-33 Web Services integration

DB2 UDB acting as a Web Service provider simply means exposing DB2 UDB data through the Web Services interface. With DB2 Information Integrator, non-DB2 data can also be exposed through the Web Services interface. Traditionally, database application developers build custom applications and wrap the applications with the Web Services interface. DB2 UDB provides tools that are integrated into the WebSphere product suite to ease the effort of creating Web Services using DB2 UDB data (see Figure 5-34).

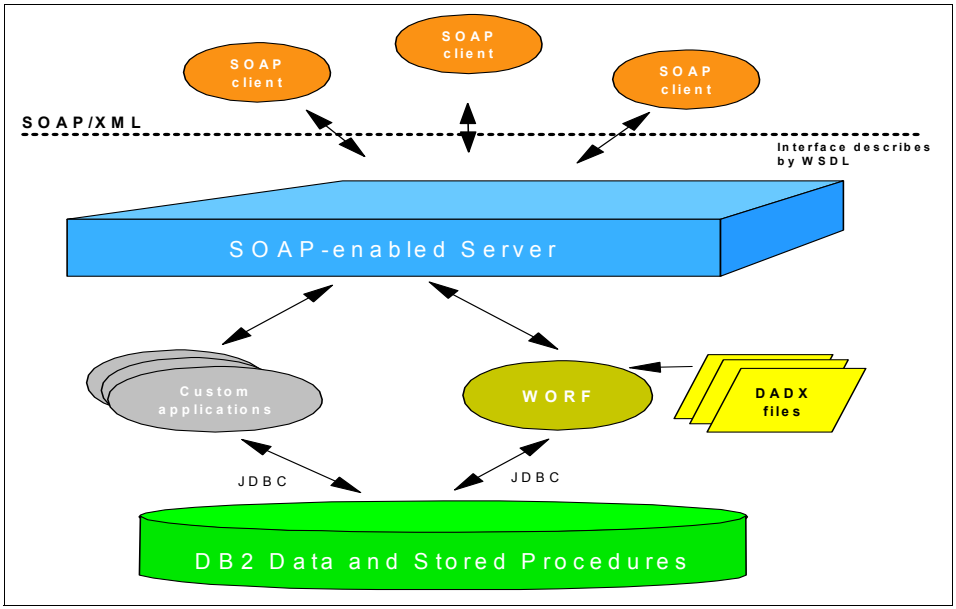


Figure 5-34 DB2 UDB as Web Services provider: implementation options

Whether custom Web Services applications or Web Services Object Runtime Framework (WORF) generated, Web Services SOAP clients are using the same interface and accessing the same data that is stored in the database.

Before we start to build DB2 Web Services, we have to introduce the technologies and tools available in DB2 UDB that enable us to easily build Web Services. First, we introduce the Web Services Object Runtime Framework (WORF). Second, we present an overview of the Document Access Definition Extension (DADX), which is a document where we specify DB2 Web Services.

5.6.6 Web Services Object Runtime Framework

The Web Services Object Runtime Framework (WORF) provides an environment to create simple XML-based Web Services that access DB2 UDB information. WORF uses Apache SOAP 2.2 or later and the Document Access Definition Extension (DADX). A DADX document specifies a Web Service using a set of operations that are defined by SQL statements or XML Extender Document Access Definition (DAD) documents.

Web Services, or functions invoked over the Internet, specified in a DADX file are called DADX Web Services, also referred to as DB2 Web Services.

WORF features

WORF provides the following features:

- Resource-based deployment and invocations, using DADX and, optionally, other resources that help define the Web Service.
- Automatic service redeployment during development time. Changing resource definitions is reflected immediately without any server restart.
- **`HTTP GET`** and **`POST`** bindings, in addition to binding to SOAP.
- Automatic WSDL and XSD generation, including support for UDDI Best Practices.
- Automatic documentation and test page generation.

WORF supports resource-based deployment of Web Services. Resource-based deployment means that Web Services are defined in a resource file that is placed in a directory of the Web application.

When that resource file is requested, WORF loads it and makes it available as a Web Service. In cases where the definitions (operations) are modified in the resource file and the same resource is requested again, WORF detects the changes and loads the modified version automatically. Automatic reloading makes DB2 Web Services development efficient and productive.

WORF generates a Web Services test client as a Web application, using Java servlet and JSPs. The test client can use simple HTTP or SOAP bindings. An HTTP binding is useful for testing a DB2 Web Service directly using a Web browser. The SOAP binding can be used by Web Services clients to create distributed applications.

WSDL is a general purpose XML language for describing the interface, protocol bindings and the deployment details of network services. WSDL complements the UDDI standard by providing a uniform way of describing the abstract interface and protocol bindings of arbitrary network services. UDDI best practices is an attempt to clarify the relationship between the two (WSDL and UDDI) and describe how WSDL can be used to help create UDDI business service descriptions.

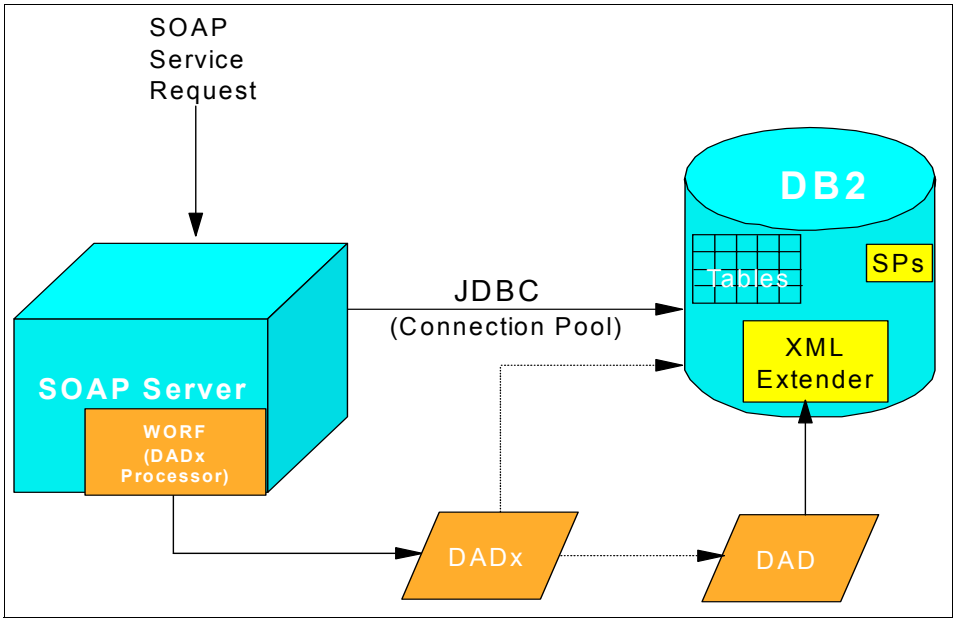


Figure 5-35 WORF architecture

Figure 5-35 shows how WORF uses the different components when processing a Web Service request. WORF receives an HTTP, SOAP, GET, or POST service request. The URL of the Web Service request includes the name of the Web Service's resource file and a command. The command is either a built-in command or an operation of Web Service specified in the resource file. The built-in commands are: TEST, WSDL, and XSD (XML schema file). Upon receiving the command, WORF generates an HTML test page for TEST, a WSDL document for WSDL, or an XML schema file for XSD.

In case the command is an operation, WORF invokes the specified operation of the Web Service and returns the result document. WORF performs the following steps to handle a Web Service request:

1. Load the DADX file specified in the request.
2. Generate a response, based on the request:
   a. For operations:
      - Load the associated DAD file, if required.
      - Replace query parameters with requested values.
      - Connect to DB2 UDB and runs any SQL statements, including SQL calls.
      - Commit the database transaction.
      - Format the result into XML, converting types as necessary.

b. For commands:
   - Generate necessary files, test pages, or other responses required.

3. Return the response to the service requestor.

WORF supported environment

WORF is available on Windows® NT, Windows 2000, AIX, Solaris, and Linux. WORF is available from the DB2 XML Extender Web site, or with DB2 UDB Version 8 and WebSphere Studio Version 4 and Version 5. When delivered with WebSphere Studio, WORF is supported with a set of tools that automate the building of DADX Web Services. These tools include a wizard to create DADX files based on SQL statements or DAD files, and tools to create DAD files. WORF also works with Informix.

5.6.7 DADX overview and structure

A DADX file specifies how to create a Web Service using a set of operations that are defined by SQL statements (including stored procedure calls) and, optionally, DAD files if the Web Services store and retrieve XML documents managed by DB2 XML Extender. WORF provides the run-time support for invoking DADX files as Web Services in the Apache SOAP 2.2 (or later) engine supported by IBM WebSphere Application Server and Apache Jakarta Tomcat.

WORF allows for the specification of storage and retrieval operations on XML data. It also allows for the exposure of stored procedures and SQL statements as invokable Web Service operations.

One can expose any database stored procedure as long it has result sets with fixed metadata (a fixed number and a fixed shape). The operation signature includes the input and output parameters. One can also specify SQL statements to select, insert, update, and delete data. Simple mapping of XML schema to SQL data type is provided.

Exposing SQL statements and stored procedures do not require the XML Extender to be installed and active.

DADX files support two kinds of Web Services operations:

- **XML collection operations (requires DB2 XML Extender):** These are storage and retrieval operations that help map XML document structures to DB2 UDB tables so that XML documents can be composed from existing DB2 data, or XML documents can be decomposed into DB2 data. This method is useful for data documents that are frequently updated.

  There are two elements that make up the XML collection operations:

  <retrieveXML> generates XML documents.

<storeXML> stores XML documents.

The DAD file provides the mapping of XML documents to a DB2 UDB database for both storage and retrieval operations.

- **SQL operations:** SQL-based querying is the ability to execute SQL statements, including stored procedure calls, to DB2 UDB and to return results with a default tagging. The data is returned using only a simple mapping of SQL data types, using column names as elements. There are three elements that make up the SQL operations:

  <query> queries the database.

  <update> inserts into, deletes from, or updates a database.

  <call> calls stored procedures with multiple result sets.

A copy of the XML schemas of DADX (DADX.xsd) is shipped with WORF. The file is located inside the <worf_unzipped_dir>\schemas directory.

The industry is moving towards the service oriented architecture. IBM WebSphere business integration leverages SOA to provide a common programming model and utilizes Web Services as the enabling technology to access information and to integrate applications.

Among its many benefits, Web Services:

- Promotes new levels of reuse
- Improves time to market
- Facilitates new levels of integration
- Unifies disparate computing paradigms
- Links diverse underlying technologies
- Provides consistent business/technical vocabulary
- Spans the entire software development cycle
- Crosses enterprise boundaries

Through DB2 UDB and Web Services integration, we can truly provide universal access to the DB2 UDB data.

5.7 Glimpse of the future

In the future, we will see many interesting updates in the technologies discussed in this chapter.

5.7.1 Lotus Workplace

Today, Lotus Workplace enables collaboration and communication between people. In the future, Lotus Workplace will also include workflow capabilities to run human-oriented business processes.

Users will have tools to design their business processes. These tools won't need developer or technical skills and will be end-user oriented.

As a result, a user will be able to manage, in their own workplace, the workflow of the different human activities associated with their role in the business and collaborate with other people in their team during the different steps of the business process.

By integrating collaboration and human interactions with processes and information, an on demand business will have the capacity to react quickly to changing business needs.

5.7.2 Information Integrator updates

- In October 2003, IBM announced the acquisition of mainframe-centric data access and integration software from CrossAccess. It will dramatically accelerate the integration of mainframe data, such as IMS/DB, VSAM, and DB2, and minimizes the dependence on mainframe skills and programming.
- Integration with other packaged applications like SAP, Siebel, and Peoplesoft will be provided.
- In future versions, the performance, scalability, and robustness of Information Integrator will be highly improved with the introduction of federated query parallelism.
- A new function will enable the discovery and modeling of data. It will allow for the definition of Integration Views at the level of the enterprise. These Integration Views are necessary to integrate the data sources of the enterprise, in the same way WBI Generic Business Objects are used to integrate different applications. It means that one can define metadata at the level of the enterprise to use a common definition of data across applications. The metadata will be stored in an XML Metadata Registry.

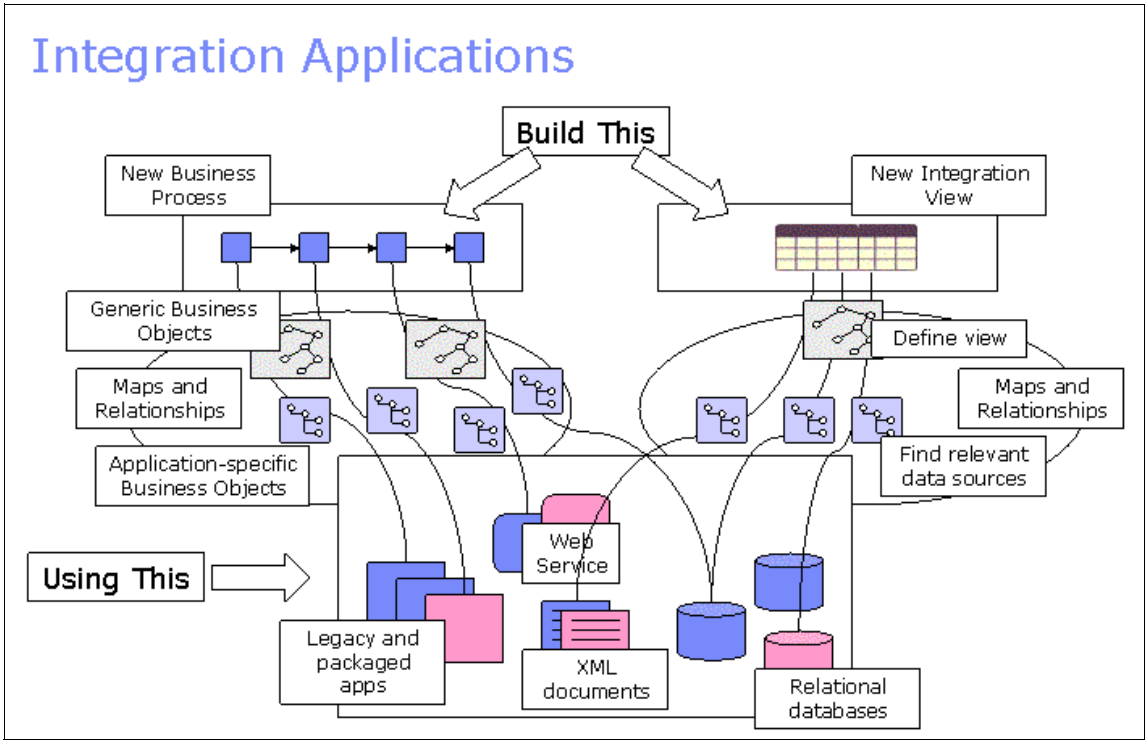


Figure 5-36 Integration views

- The delivery of an XML Metadata Registry will also be a major step forward. Metadata management is a pervasive problem. Most software products and development tools today generate some sort of metadata (EMF, UML, XML Schema, WSDL, BPEL). Because there is no common framework for storing metadata, this metadata is dispersed in file systems, databases, and hidden in the products themselves. As a result, customers have a difficult time knowing what metadata is out there and how it might be related. At the heart of the system will be a repository that provides mechanisms to store metadata assets (such as XML schemas, message definitions and relational table definitions), define models that convey relationships between assets, search, and exploration. This repository will provide a single view by which to manage and explore the artifacts that are generated by different groups over time. Surrounding the core will be services that allow users to discover new assets, discover implicit relationships between assets, define maps between assets, and generate run-time artifacts such as views, queries and insert statements that allow a user to combine data from heterogeneous sources. WSAD will have a plug-in to import/export XML artifacts and XML schemas.

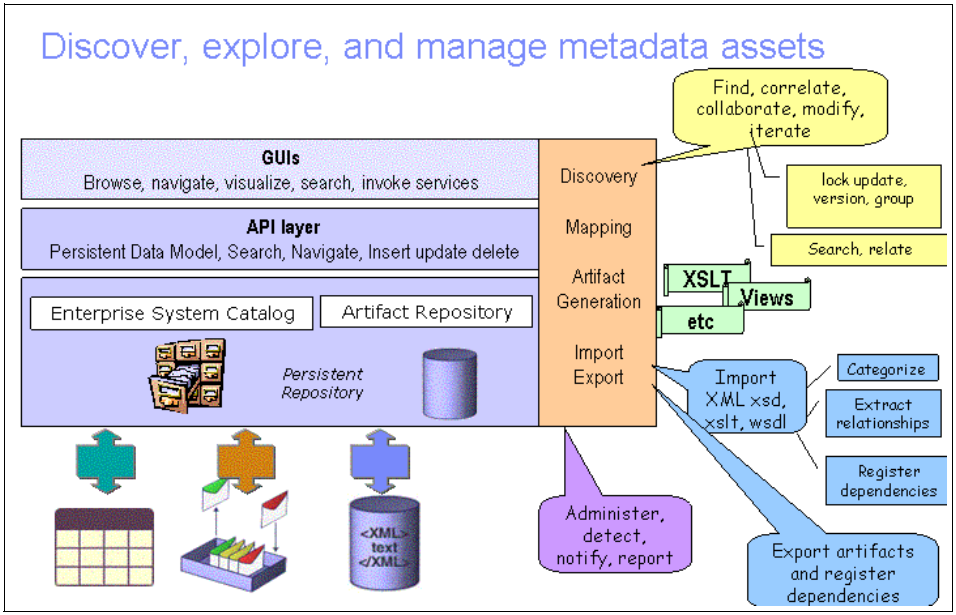


Figure 5-37 XML Metadata registry

Toolset for information integration

While the XDE toolset provides the capability to generate the physical data model for traditional databases, this capability is not currently available for “federated” data stores. XDE does not support the “nicknames” for external data stores, and therefore cannot be used for setting up the physical model in a federated environment.

In the future, there will be the capability to define federated data stores and nicknames with the XDE toolset.

In addition, there will be the capability to reverse engineer all data that is to be federated. That is, there will be the ability to obtain the metadata model from federated data sources.

Figure 5-38 illustrates this reverse engineering capability. In this case, there is already a federated database with nicknames and metadata, and the data architects want to work with logical data models. The XDE toolset will provide the reverse engineering function to build the logical data models from the federated database.

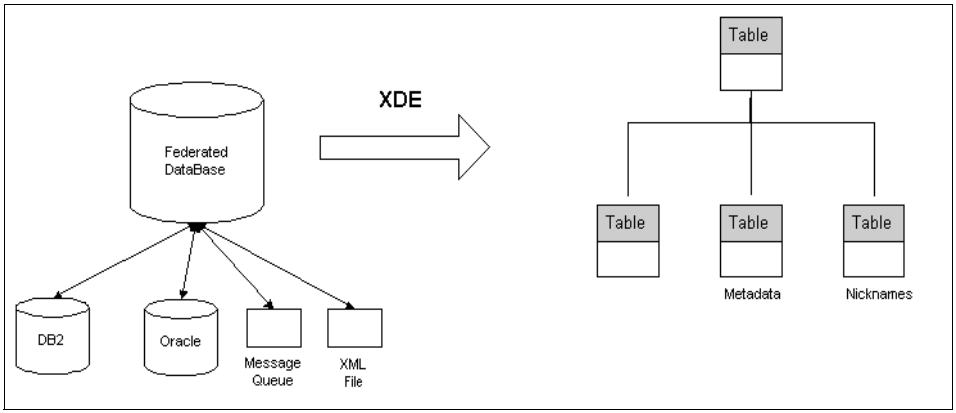


Figure 5-38 Reverse Engineering Step 1

This capability is expected to be provided in an evolutionary manner. In the first step it is envisioned to have the capability to define a metadata model for the overall federated structure through the nicknames, as shown in Figure 5-38. In the second step, reverse engineering is expected to allow creation of individual, distinct metadata models for each of the federated sources. The metadata and data models can then be refined and reorganized for any future business objective to yield more effective information federation. The data architect uses the logical data models to define the nicknames and the metadata of the federated database. This is shown in Figure 5-39.

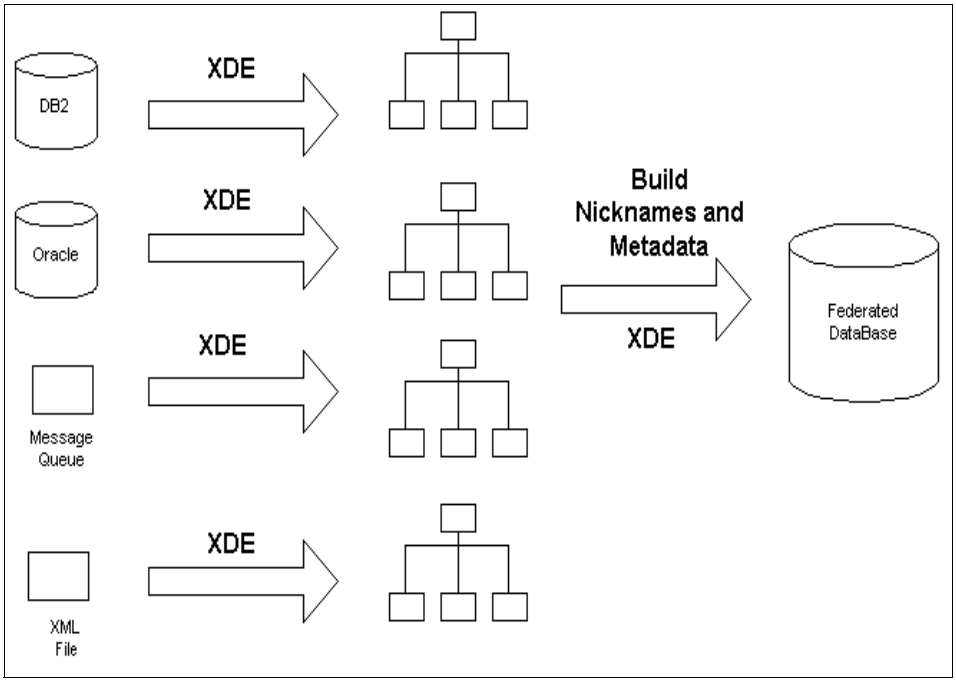


Figure 5-39 Reorganized federated information based on reverse engineering

5.8 Summary

This chapter discussed how customers can react in near real time to most relevant information by ensuring a seamless flow of information in the extended enterprise through the on demand Operating Environment. It discussed the elements of the Operating Environment framework that apply to this class of business problem, and how it allows a business to achieve rapid time to value.

The on demand Operating Environment allows information from various sources within the enterprise and outside it from business partners to be integrated and made available in near real time to those requiring it. The Operating Environment also provides the foundation for presentation of this information in a personalized manner and for allowing various people to collaborate to make decisions that affect the business objectives. It is set up to allow an evolutionary path while preserving investment in existing infrastructure, skills, and applications.

The elements of the on demand Operating Environment that applied were discussed in the context of a business scenario that showed how customers

could apply these concepts, in an evolutionary manner, preserving their investment in tools, skills, and legacy systems.

The tools and infrastructure discussed in this chapter provide significant value today. The longer term vision is to enhance them to provide greater insight into the various information elements in the extended enterprise and how they can be used to provide business value.

A

Standards overview

An on demand computing environment involves diverse systems working together and connecting to devices and applications across platforms, organizations, and even geographic borders. This environment helps to enable a business to respond quickly to changes in markets, technologies, and the needs of their customers. Businesses must be able to rapidly provide new capabilities to their systems without completely discarding and replacing those applications and systems.

The only way all of these components, applications and systems can work together is with open industry standards. With open standards, businesses and providers can ensure that their products and systems will work and communicate with other systems.

Standards enable an on demand business to create the environment and applications needed. IBM is actively involved in developing many of these standards, as are many other companies worldwide.

Open source

Open standards are necessary to open-source projects. Open-source projects frequently provide implementations of key standards that serve as references. The Apache Web server is one example. Because the Apache Web server is so commonly used, every browser must work with its implementation of the

Hypertext Transfer Protocol (HTTP) standard. This creates a market pressure that prevents vendors from introducing incompatible, proprietary implementations of HTTP.

IBM contributes to open-source projects, and actively supports the open-source community. IBM contributed an Extensible Markup Language (XML) parser to the Apache Xerces project, and an XML Stylesheet Language Transformation (XSLT) engine to the Xalan project. In addition, IBM created the Eclipse project, an effort to create an open-source integrated development environment (IDE). In turn, many open-source tools are being incorporated into IBM development tools.

Standards organizations

There are many standards organizations contributing to the key standards for on demand computing. The following are some of the key standards organizations that one should be aware of when planning for an on demand Operating Environment.

- Internet Engineering Task Force (IETF)
- World Wide Web Consortium (W3C)
- Java Community Process (JCP)
- Organization for the Advancement of Structured Information Standards (OASIS)
- Web Services Interoperability Organization (WS-I)
- Distributed Management Task Force (DMTF)
- Global Grid Forum (GGF)
- Object Management Group (OMG)

IETF - Internet Engineering Task Force

The Internet Engineering Task Force (IETF) creates standards for the operation of the Internet infrastructure. The IETF was formed in 1986 and has evolved into an active standards organization involving thousands of people from academia, research, and industry. The IETF has no formal membership. Anyone may participate in mailing lists or attend meetings. Participants are organized into an ever-changing collection of Working Groups, which are further organized into Areas. While IETF Working Groups can be created in any area based on the interests of the participants, in practice, the IETF concentrated on the transmission of Internet Protocol (IP) packets and the information required to secure, route, and manage the communications.

Because on demand computing assumes computer networking as a base capability, almost all IETF standards have an impact. A unique requirement of on demand computing, though, is highly scalable and dynamic networking. The IETF is leading the transition of the Internet infrastructure to a new base protocol, known as IPv6, which will dramatically increase the number of Internet addresses available and simplify address management. In addition, the IETF continues to evolve security and routing protocols that enable dynamic creation of secure networks.

W3C - World Wide Web Consortium

The World Wide Web Consortium (W3C) creates specifications for Web technologies. The mission of W3C is to lead the Web to its full potential. It does this by developing recommendations, guidelines, software and tools that create a forum for information, commerce, inspiration, independent thought, and collective understanding.

Tim Berners-Lee, who is widely credited as being the architect of the World Wide Web, founded the W3C to further the growth of the Internet and ensure its interoperability. The W3C is a consortium of companies working together to develop Web technologies. HTML, Extensible Hypertext Markup Language (XHTML), XML and XML Schema are just a few examples of W3C Recommendations.

The W3C organizes the work on the development of Web technologies into Activities. The structures of these Activities vary, but each Activity usually includes a Working Group, an Interest Group, and a Coordination Group. Within their respective Activities, these groups produce Recommendations and other technical reports as well as sample code.

W3C activities of interest include the following:

- XML family of standards (the XML, XML Base, XML Query, XML Schema, XPath, and XSLT standards are of particular interest)
- Simple Object Access Protocol (SOAP)
- Web Services Description Language (WSDL)

JCP - Java Community Process

The Java Community Process (JCP) organization, created by Sun Microsystems, was formed to create and maintain Java technical specifications. The companies and Java developers who make up the JCP provide specifications, reference implementations, and technology compatibility kits to guide the evolution of Java technology. The open organization works with

member and nonmember input. Anyone can join the organization, but membership is not required in order to contribute.

The JCP works to ensure the stability and cross-platform compatibility of Java. It also works to expand the platforms specification portfolio to address emerging technologies.

OASIS - Organization for the Advancement of Structured Information Standards

In 1993, a consortium of vendors and users formed SGML Open to develop guidelines for interoperability among products that support the Standard Generalized Markup Language (SGML). By 1998, the scope of the work had expanded to include XML and other related standards, and the name was change to the Organization for the Advancement of Structured Information Standards (OASIS).

Today, OASIS is a not-for-profit, global consortium driving the development, convergence, and adoption of e-business standards. OASIS develops structured information standards for security, Web Services, XML conformance, business transactions, electronic publishing, topic maps, and interoperability within and between marketplaces. OASIS members set the technical agenda, following a process designed to achieve industry consensus and unite disparate efforts.

Key OASIS specifications of interest to the IBM community include:

- UDDI
- Web Services Security (WS-Security)
- Business Process Execution Language (BPEL4WS)

WS-I - Web Services Interoperability Organization

The Web Services Interoperability Organization (WS-I) promotes interoperability between Web Services across platforms, applications, and programming languages. The organization includes diverse group of software vendors, enterprise customers, and others interested in Web Services to aid the development of interoperable Web Services with guidance, recommended practices, and resources. The WS-I provides resources to help developers create Web Services that are interoperable and compliant with WS-I guidelines and industry standards.

The WS-I Basic Profile 1.0 specification describes ways in which diverse Web Services specifications can work together to create interoperable Web Services.

The WS-I is also working on a profile to cover the implications and workings of the OASIS standard, WS-Security.

DMTF - Distributed Management Task Force

The Distributed Management Task Force, Inc. (DMTF) promotes the development and adoption of interoperable management standards for enterprise and Internet environments. They developed the Common Information Model (CIM) standard, which describes a platform-independent method for exchanging management information. The standard helps to simplify integration and reduce costs for management systems by enabling an end-to-end multi-vendor interoperability. By implementing CIM, vendors and standards groups make possible more integrated and cost-effective management systems.

DMTF standards with which you should be familiar for on demand computing are:

- CIM
- Web Based Enterprise Management (WBEM)

GGF - Global Grid Forum

The Global Grid Forum (GGF) promotes and supports Grid technologies and applications. They create specifications, user experience documents and implementation guidelines to help organizations developing, deploying and implementing Grid technologies.

In addition, they promote the development of a broad-based Integrated Grid Architecture to support emerging Grid communities. Such architecture can aid the Grid agenda by spreading necessary basic services and encouraging the use of shared middleware code for applications with common requirements.

GGF recommendations with which you should be familiar for on demand computing include:

- Open Grid Services Infrastructure (OGSI), a base set of distributed computing operations to support dynamic middleware
- Open Grid Services Architecture (OGSA), a model of a computing system as a set of distributed computing patterns realized as applications and extensions of Web Services
- Distributed Resource Management Application API (DRMAA), an application programming interface specification for the submission and control of jobs to one or more Distributed Resource Management (DRM) systems

OMG - Object Management Group

The Object Management Group (OMG) is a nonprofit consortium whose purpose is to promote object-oriented technology and the standardization of that technology. The OMG was formed to help reduce the complexity, lower the costs, and accelerate the introduction of new software applications. Some of OMG's major successes include the Common Object Request Broker Architecture (CORBA), and the Unified Modeling Language (UML). One of OMG's current efforts is establishing the standards for Model-Driven Architecture (MDA).

OMG specifications with which you should be familiar for on demand computing include:

- MDA
- UML

Although W3, OASIS, IETF and OMG are key standards-setting bodies for our future Grid services world, it is important for developers to follow the interoperability standards set by the WS-I. Web Services support and make possible key elements of the emerging Grid services.

Key standards

At least 158 standards are significant to the on demand strategy. These standards apply to any of 22 different categories, including messaging, security, management, Java, and discovery categories, to name just a few.

There are also "vertical" standards that support the on demand strategy. Vertical standards refer to business standards or regulations that developers must follow when developing software for particular sectors or industries. Examples of these vertical standards include RosettaNet for electronics and ACORD for insurance.

The following are some of the key standards that apply to an on demand Operating Environment:

- XML standards, including XML Schema and XSLT
- SOAP
- WSDL
- UDDI
- WS-I Basic Profile
- WS-Security
- OGSA

- OGSI
- UML
- MDA
- WBEM
- CIM, CIM-XML, CIM-SOAP

Of these, SOAP, WSDL, UDDI, WS-I Basic Profile, and WS-Security are basic Web Services standards. Figure A-1 shows how these and other Web Services standards relate to one another.

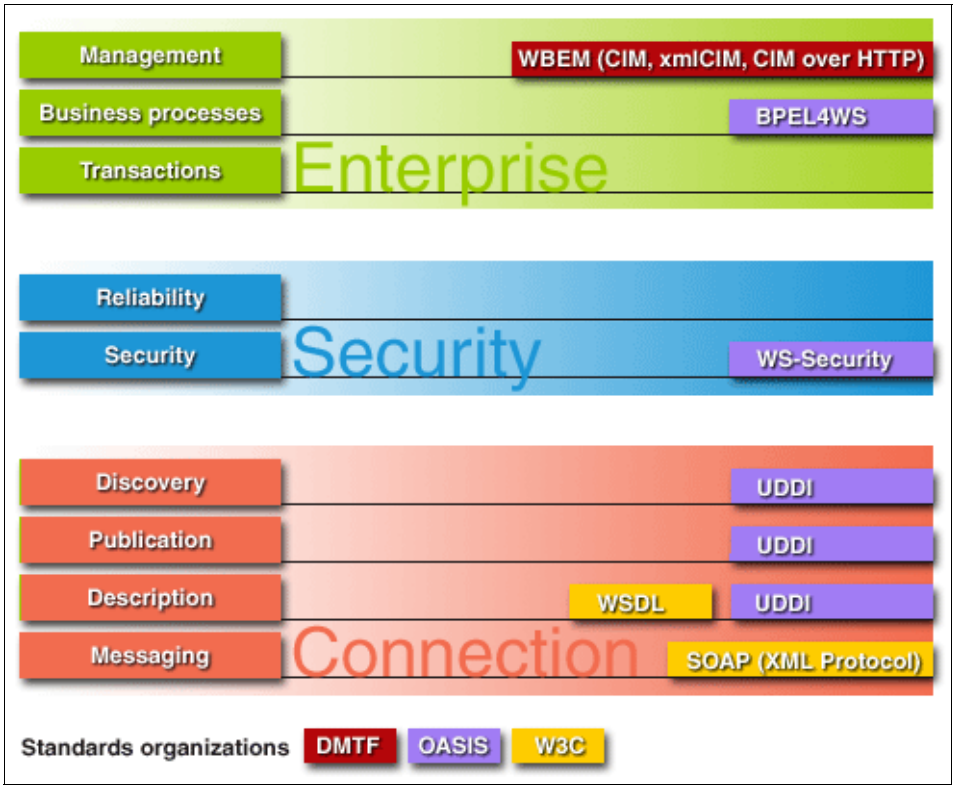


Figure A-1 Key standards

XML standards

XML is a family of technologies for defining and processing application-specific markup languages that describe data and documents. Specifically, XML is a language for creating and using structured information. XML is based on, and is a subset of, SGML.

In simple terms, XML is simply a standardized format for the representation of data documents. It was developed by an XML Working Group formed under the auspices of the W3C in 1996 and provides the foundation for many of the open standards of today. This is particularly true of those standards related to the interoperability of applications and components, such as WSDL, since XML defines a simple base structure for the representation of data.

Resources on the details of XML syntax and related technologies are numerous.

Some of the inherent benefits of XML include:

- It is an easy-to-use, open standard for data description and as such, forms a convenient common ground between heterogeneous applications and components.
- Its element-oriented structure means that XML is indeed easily extensible. A common problem with proprietary file formats (such as fixed-width record files) is that they are often only able to withstand a finite amount of extension (lack of space in the record, for example). The tag structure of XML makes the addition of new tags and attributes straightforward.
- XML documents are, generally speaking, easier for humans to read and understand (and therefore, debug or analyze) than comma-separated or hash-delimited files. For example, compare the following data formats, which relate to the same piece of data:

  In a hash-delimited format:

  ```
  1#martin#gale#d0168
  ```

  In XML:

  ```
  <employee id="1">
  <name><forename>martin</forename><surname>gale</surname></name>
  <office>d0168</office>
  </employee>
  ```

- XML defines languages for describing the structure of a particular XML document in order for it to be valid in terms of its application. XML standards describe the syntactical constructs for the base markup of a document. The validation uses a Document Type Definition (DTD) document or, more recently, an XML Schema document, both of which describe the validation rules for the data. DTDs and XML Schemas are referenced from within a given XML document using a Uniform Resource Locator (URL). This allows the document to be validated regardless of the platform on which it is processed.
- A variety of freely available, open-source XML parsers for various programming languages make integrating structured data described in XML into an application straightforward. Likewise, the availability of XSLT

processors means that the translation of XML from one format into another is a portable and straightforward process.

XML Schema

XML Schema is a key XML-related technology in the on demand world. XML Schemas express shared vocabularies and allow machines to carry out rules made by people. They provide a means for defining the structure, content and semantics of XML documents, and allow the entities within the XML data to be bound to user-defined type information. Schema is a modernization of the XML DTD principle that is in itself described in XML, as opposed to DTD's proprietary format.

As an XML vocabulary itself, Schema carries with it all the benefits of XML, particularly in respect to portability. In this way, XML Schema documents are used in conjunction with WSDL in Web Services to describe not only the Web Service provided, but to define the data types consumed by that service. Similarly, XML-based standards are now appearing defined in XML Schema. In these cases, the schema provides a single standard format artifact that describes the vocabulary in question, and is a valid run-time asset with which to validate the documents using it.

XSLT

Other key standards conveyed in XML are XSLT. In general terms, XSLT stylesheets describe the mapping of an XML document from one XML format to another. XSLT stylesheets can transform a document containing data into output markup in which the data is contained within formatting constructs (such as XHTML). In addition, where a key standard is expressed as an XML vocabulary, you can also use XSLT itself to generate new controlling documents, including XSLT stylesheets and XML Schema definitions. In many ways, XSLT is useful as a flexible, portable, and relatively easy to use markup.

SOAP - Simple Object Access Protocol

SOAP is an XML-based protocol for applications to send and receive messages over the Internet. It is a W3C Recommendation. SOAP defines an "envelope" that allows clients and service providers to communicate and exchange XML-formatted data, regardless of platform or programming language. The specification defines the XML formatting for the messages, a method for encoding the data as XML and a binding to HTTP as the transmission method.

The specification allows for using other encoding methods, but this is discouraged because it would limit the potential users and potentially fragment

the SOAP user base. The SOAP specification also allows other bindings, such as WebSphere MQ.

WSDL - Web Services Description Language

WSDL is an XML format for describing the interfaces of a Web Service. The details described include the protocols and port numbers used, available operations, and message formats. WSDL is being defined by the W3C. In basic terms, the language is used to describe the capabilities of a Web Service (for example, the operations that can be performed). The details described include the protocols and port numbers used, message formats and possible exceptions.

IBM and Microsoft jointly developed WSDL. Note that the UDDI standard uses WSDL.

UDDI - Universal Description, Discovery, and Integration

UDDI is a business-registry specification used to support the description and discovery of Web Services providers, including businesses and organizations, the Web Services offered by those providers, and the interfaces available to access those services. UDDI is an OASIS specification for indexing Web Services so that users can locate and use them.

UDDI makes use of several standards, including SOAP, XML Schema, and WSDL. It provides a mechanism for clients to find other Web Services. Entries in a UDDI registry include information on the business offering a Web Service, the capabilities of the services offered, and technical details on how to invoke and use the service.

WS-I Basic Profile 1.0a

The WS-I Basic Profile Version 1.0a was released August 8, 2003. The Basic Profile describes a manner in which four key Web Services specifications can be implemented to achieve a consistent measure of interoperability. Those four specifications are:

- XML Schema 1.0
- SOAP 1.1
- WSDL 1.1
- UDDI 2.0

Other standards organizations manage these specifications. The Basic Profile does not add to the specifications. It seeks to show how they can work together.

Among other issues, the Basic Profile addresses the major interoperability concerns, provides a way of testing, clarifies the requirements in specifications (such as avoiding optional components as sources of possible confusion), and makes strong recommendations regarding multiple possible implementation mechanisms that may be found in the specifications themselves.

WS-Security - Web Services Security

Web Services Security (WS-Security) is a proposed specification for enhancing the security of SOAP messaging. Developed jointly by IBM, Microsoft, and Verisign, WS-Security has been submitted to OASIS.

WS-Security provides a basic mechanism for linking security tokens to SOAP messages. Designed to support multiple types of security tokens, it does not specify the use of any particular one. WS-Security describes how the binary tokens should be encoded. By itself, WS-Security does not ensure security, but it is designed to make use of other Web Services extensions and higher level application-specific protocols.

OGSA - Open Grid Services Architecture

The OGSA is a model of a computing system as a set of distributed computing patterns realized as applications and extensions of Web Services. IBM supports the development of OGSA for the management of a virtualized set of resources. The Global Grid Forum manages the OGSA effort.

OGSA defines the elements necessary to build and run a platform for distributed system integration. These elements include:

- The scope of the services needed to support scientific and business applications
- The core set of services necessary for Grid systems and applications
- The functions needed for the core services and the relationships between them

OGSA integrates key Grid technologies with Web Services mechanisms to create a distributed system framework based on the Open Grid Services Infrastructure (OGSI). Unlike OGSI, OGSA addresses the creation, management and destruction of Grid services.

OGSI - Open Grid Services Infrastructure

OGSI defines methods for the creation, management, and exchange of information between Grid services. A Grid service is defined as a Web Service

that conforms to a set of conventions, expressed as WSDL interfaces, extensions, and behaviors. The elements address purposes such as lifetime management, discovery of characteristics, and notification. Grid services provide a method for managing the distributed and possibly prolonged state that is often required by complex distributed applications. OGSI also specifies methods for the creation and discovery of Grid services with standard factory and registration interfaces.

According to the current draft of the specification, the OGSI component model extends WSDL and XML Schema to incorporate:

- Stateful Web Services
- Inheritance of Web Service interfaces
- Asynchronous notification of state change
- References to instances of services
- Collections of service instances
- Service state data that augments the constraint capabilities of XML Schema definition

Note that OGSI defines the basic elements and mechanisms that OGSA uses to create a Grid services platform.

DRMAA is a related specification being developed by the GGF for the submission and control of jobs to one or more DRM systems.

Figure A-2 illustrates how the Grid specifications relate to other standards. Grid computing makes extensive use of Web Services. In turn, Web Services rely on XML. None of it would be practical without the underpinnings of the Internet and the programming that creates the applications involved.

Figure A-2 Grid specifications and related standards

UML - Unified Modeling Language

UML is a standard notation used to visually design and model applications and systems. UML is a language and not a methodology, and as such, it is independent of programming languages or platforms. It can be used with any programming language to create design plans and illustrate system structures. UML diagrams and documents essentially serve as blueprints. Although UML is used extensively in application development, it can also be used for business modeling and other non-software types of systems. UML is an OMG specification and forms the basis of another OMG specification, MDA.

MDA - Model Driven Architecture

MDA is an OMG specification based on UML. Making extensive use of UML models, MDA offers a way of creating specifications and developing applications

separated from the underlying technologies of the platforms used. Leveraging UML, developers using MDA can design interfaces and relationships between applications independent of platforms and programming languages. The applications designed this way can be created on a range of platforms, open or proprietary. When new technology is developed, the modeling does not need to be repeated.

The Eclipse project is an example of an industry open model framework that is MDA compliant.

Using MDA, you start with a Platform Independent Model (PIM). This is a model of functionality and behaviors without details on the technical implementation. MDA-compliant tools then map the PIM to a Platform-Specific Model (PSM), or more likely, to multiple PSMs. This partially automatic process is accomplished using OMG-standardized mappings. The resulting PSMs are also UML models.

The MDA tools then use the PSM models to generate actual code, including interfaces, configuration files, and more. Depending on the complexity of the model or application, the tools generate all or most of the code needed. At this point, the code can be fine-tuned before the application is deployed.

CIM - Common Information Model

CIM is a DMTF standard for expressing data about systems, applications, networks, and devices. CIM allows various management applications to access the data and control the devices or systems regardless of the platforms involved, making interoperability easier to achieve.

The CIM provides object classes, properties, methods, and associations common to the use of management applications in the form of management schema. These are organized into three layers:

- A core model addresses elements that span all areas of management.
- Common models address elements found in specific management areas, such as systems, applications, networks, and devices, independent of technologies or implementations.
- Extension schemas address the needs of specific technologies (specific operating systems or platforms).

CIM is independent of the method used for implementation.

Web-Based Enterprise Management (WBEM) Initiative

The Web-Based Enterprise Management (WBEM) Initiative is an effort by the DMTF to design standards for the management of computing environments. The

goal is to provide the standards that the industry can use to create integrated, standards-based tools that make use of Web technologies. At the core of this initiative is the CIM.

Related standards include:

- xmlCIM specifies a way in which CIM elements and messages can be expressed in XML.
- CIM Operations over HTTP specifies how to map CIM operations onto HTTP to achieve an open, standardized interoperability.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 228. Note that some of the documents referenced here may be available in softcopy only.

- *IBM Informix: Integration Through Data Federation,* SG24-7032
- *Patterns: Direct Connections for Intra- and Inter-enterprise,* SG24-6933
- *Getting started on Integrating your information*, SG24-6892

Online resources

These Web sites and URLs are also relevant as further information sources:

- IBM On demand web pages:

  `http://www.ibm.com/ondemand`
- WebSphere redbooks

  `http://publib-b.boulder.ibm.com/redbooks.nsf/portals/WebSphereRedbooks`
- IBM Software

  `http://www.ibm.com/software`
- Rational tools example

  `http://www.ibm.com/developerworks/rational/library/184.html`
- WebSphere Business Integration Adapters

  `http://www.ibm.com/software/integration/mqfamily/adapter`
- WebSphere Business Integration Connect

  http://www.ibm.com/software/integration/wbiconnect
- Rational Developer Network

  `http://www.ibm.com/software/awdtools/rup/index.html`

- IBM DeveloperWorks

  www.ibm.com/developerworks
- World Wide Web Consortium

  www.w3c.org
- OASIS organization

  www.oasis-open.org

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

D

E

F

G

H

I

J

Q

R

S

T

U

V

W

X

IBM Redbooks

On demand Operating Environment: Creating Business Flexibility

(0.2”spine)
0.17”<->0.473”
90<->249 pages

On demand Operating Environment: Creating Business Flexibility

Introducing the on demand Operating Environment

Enabling integration between people, processes, and data

Start deploying today

This IBM Redbook (along with its companion volume *On demand Operating Environment: Managing the Infrastructure*, SG24-6634), provides an insight into the kind of operating environment required to support an on demand business.

It provides an overview of the architecture of an on demand Operating Environment and describes in detail the components that are required to create business flexibility through integration. To meet the business needs of being responsive, variable, focused, and resilient, an on demand Operating Environment must be integrated, autonomic, virtualized, and open. Though these attributes are all interelated, this redbook focuses on the integration component as it enables business flexibility.

This redbook provides descriptions of several approaches that one can choose to start implementing pieces of an on demand Operating Environment today. Which approach is right for the reader will depend on their specific business environment and their immediate needs.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6633-00 ISBN 0738498262